2

THE ANALYTIC SCIENCES CORPORATION

TR-9033-2

TASK ORDER 33

SOFTWARE MEASUREMENT PROCESS

Evaluation of Software Measurement Processes and Software Measurement Support

Technical Report

3 May 1989

Prepared Under:

Contract No SDIO84-88-C-0018

Prepared For:

STRATEGIC DEFENSE INITIATIVE ORGANIZATION
Office of the Secretary of Defense
Washington, D.C. 20301-7100

Prepared By:

SPARTA, Inc.
Teledyne Brown Engineering
The Analytic Sciences Corporation

THE ANALYTIC SCIENCES CORPORATION
1700 North Moore Street
Suite 1800
Arlington, VA 22209

UNCLASSIFIED
SECURITY CLASSIFICATION OF THIS PAGE

REPORT DOCUMENTATION PAGE

Form Approved
OMB No. 0704-0188

1a. REPORT SECURITY CLASSIFICATION
UNCLASSIFIED

1b. RESTRICTIVE MARKINGS

2a. SECURITY CLASSIFICATION AUTHORITY

2b. DECLASSIFICATION / DOWNGRADING SCHEDULE

3. DISTRIBUTION / AVAILABILITY OF REPORT
Approved for Public Release
Distribution Unlimited

4. PERFORMING ORGANIZATION REPORT NUMBER(S)
TR-9033-2

5. MONITORING ORGANIZATION REPORT NUMBER(S)

6a. NAME OF PERFORMING ORGANIZATION
The Analytic Sciences
Corporation (Prime Contractor)

6b. OFFICE SYMBOL *(If applicable)*

7a. NAME OF MONITORING ORGANIZATION
Strategic Defense Initiative Organization

6c. ADDRESS *(City, State, and ZIP Code)*
1700 N. Moore Street
Suite 1800
Arlington, VA 22209

7b. ADDRESS *(City, State, and ZIP Code)*
Room 1E149
The Pentagon
Washington, D.C. 20301-7100

8a. NAME OF FUNDING / SPONSORING ORGANIZATION
Strategic Defense Initiative Organization

8b. OFFICE SYMBOL *(If applicable)*

9. PROCUREMENT INSTRUMENT IDENTIFICATION NUMBER

8c. ADDRESS *(City, State, and ZIP Code)*
Room 1E149
The Pentagon
Washington, D.C. 20301-7100

10. SOURCE OF FUNDING NUMBERS

PROGRAM ELEMENT NO.	PROJECT NO.	TASK NO	WORK UNIT ACCESSION NO.

11. TITLE *(Include Security Classification)*
Evaluation of Software Measurement Processes and Software Measurement Support (U)

12. PERSONAL AUTHOR(S)

13a. TYPE OF REPORT

13b. TIME COVERED
FROM 5Dec88 TO 3May89

14. DATE OF REPORT *(Year, Month, Day)*
1989, May, 3

15. PAGE COUNT
125

16. SUPPLEMENTARY NOTATION TASC Report No. TR-9033-2
Prepared by SPARTA, Inc, Teledyne Brown Engineering, and The Analytic Sciences Corporation

17. COSATI CODES

FIELD	GROUP	SUB-GROUP

18. SUBJECT TERMS *(Continue on reverse if necessary and identify by block number)*
SDS SOFTWARE, EVALUATION, MEASUREMENT PROCESS, METRICS,
MEASUREMENT PLAN Strategic Defense Initiative Organization

19. ABSTRACT *(Continue on reverse if necessary and identify by block number)*

This report presents an evaluation of current software measurement processes and software measurement support tools. The applicability and requirements for the use of these tools to SDS softwarê development were identified. The evaluation was performed in following steps: (1) collection of extensive metrics data from industry, Government and academic sources, (2) evaluaton and analysis of the collected information for specific relevancy within ~~SDIO~~ software domain applications, (3) review of tools databases and product information literature to identify potential metrics tools applicable to SDS Software support, and (4) identification of deficiencies or domain limitations of existing methods and models.

The results of our review of available and ongoing metrics program reveal that a metrics methodology is needed for the effective use of metrics and each major system, development/acquisition must develop and tailor its own metrics program. There is a need to emphasize predictive measurands in the earlier phases of the life cycle. (KR) ←

20. DISTRIBUTION / AVAILABILITY OF ABSTRACT
☒ UNCLASSIFIED/~~UNLIMITED~~ □ SAME AS RPT. ☒ DTIC USERS

21. ABSTRACT SECURITY CLASSIFICATION
UNCLASSIFIED

22a. NAME OF RESPONSIBLE INDIVIDUAL

22b. TELEPHONE *(Include Area Code)*
(202) 693-1600

22c. OFFICE SYMBOL
SDIO/S/PI

Previous editions are obsolete.

SECURITY CLASSIFICATION OF THIS PAGE
UNCLASSIFIED

TABLE OF CONTENTS

TABLE OF CONTENTS (Continued)

LIST OF TABLES

LIST OF FIGURES

LIST OF ACRONYMS

AMMCOM	U.S. Army Armaments Command
CSC	Computer System Component
CSCI	Computer System Configuration Item
IDA	Institute for Defense Analyses
IEEE	Institute for Electrical and Electronics Engineering
ISEC	U.S. Army Information Systems Engineering Command
N-SITE	Near-Term System Integration, Test and Evaluation
RADC	Rome Air Development Center
SDC	Strategic Defense Command
SDIO	Strategic Defense Initiative Organization
SDLC	Software Development Life Cycle
SDS	Strategic Defense System
SIE	Satka (Surveillance, Acquisition, Tracking and Kill Assessment) Integrated Experiment
SOA	State of the Art
SOW	Statement of Work
SPO	System Program Office
TASQ	Technology for the Assessment of Software Quality
USNPGS	U.S. Naval Post-Graduate School
V&V	Verification and Validation

EXECUTIVE SUMMARY

OBJECTIVES

The work presented in this report was performed under Subtask 2 and 3 of Task Order 33 of the SDIO Systems SETA contract. The purpose of these subtasks was to evaluate current software measurement processes and to evaluate current software measurement support tools. Teledyne Brown Engineering was the technical lead on the subtasks with support from Sparta and TASC.

The subtasks consisted of the following activities:

- collection of extensive metrics data from industry, Government and academic sources;
- evaluation and analysis of the collected information for specific relevancy within the SDIO software domain applications;
- review of tools databases and product information literature to identify potential metrics tools applicable to SDI software support;
- identification of deficiencies or domain limitation of existing methods and models.

CONCLUSIONS

There are several major conclusions from this subtask. The results of our review of available and ongoing metrics programs reveal that a metrics methodology is needed for the effective use of metrics and each major system, development/acquisition must develop and tailor its own metrics program. SDI is no exception. There is a need to emphasize predictive measurands in the earlier phases of the life cycle (i.e., requirements and design). In the near term, available metrics tool packages can be halpful, but greater use of formal notation to support requirements and design synthesis is required if metrics information rigor and application is to occur in a much more disciplined and predictive manner.

OPEN ISSUES

The selection of application data used with metrics models must be carefully selected, scaled and scoped. Formal metrics models for use in the early life cycle phases must be developed. In turn, a life cycle model, iterative in nature, must be identified that can successfully be used for SDI development. Without a proper life cycle and metric requirements model, the state-of-the-art in predictive metrics will continue to lag behind post-facto metrics for some time to come. Predictive metrics are essential if higher productivity yields and better quality reusable software is to become a reality.

00. INTRODUCTION

This report provides an evaluation of current software measurement processes and tools currently used in the Government, industry and academia.

Section 1 lists, summarizes and analyzes the metrics models from available databases and software documents. Section 2 presents an assessment of how each applicable software metric can be applied within each SDS software subfunction. For each candidate software metric identified, a validation analysis summary will be presented. Section 3 presents the results of analyzing the field experience with two initiatives highlighted. Section 4 identifies capabilities and limitations in current methods. Section 5 presents a literature and database survey identifying existing metrics tools and environments.

The collection of metrics data proved to be both skewed and elusive. Skewed in the sense that much of the existing metrics information that is found to exist and is formalized via models and mathematical relationships is encountered late in the life cycle (i.e., occurring in the implementation phase and beyond). Elusive due to the fact that early life cycle (predictive) metrics are virtually non-existent, and when identified, have no formal foundations (mathematical or relational) to support them for the most part. Furthermore, consensus is still involing on the relative value of specific metrics (e.g., complexity) amoung metrics "experts". Essentially each program must establish and implement its own metrics and quality program to derive maximum benefit from it.

A section on life cycle models (1.6) has been incorporated into this report. This section is intended to support the need for a new life cycle model consistent with the MIL-STD-2167A requirements. It is also intended to provide further insight on the need for metrics emphasis and their relationships in the initial (early) life cycle phases, and the development of more formal predictive metrics. Such emphasis and identification appears to provide a key component to achieving higher productivity and quality thresholds.

Some of the data collected and conclusions arrived at, while disappointing due to the state-of-the-metric-art, are not surprising and confirm expectations. However, the life cycle section, together with the evidence identified in section 2.1, giving support to a multi-attribute, composite or vector metric representation presents a potential breakthrough in the SOA. The latter metrics form is the subject of the 1990 International Conference on Metrics being held in the United States for the first time in several years.

With respect to a tools/environment database, the Army's TASQ database contains a wealth of information as evidenced by Appendix C and D, and has served as an extremely valuable source of information, representing the latest and most extensive survey in the marketplace.

This document is driven by the software measurement requirements established in the Task Order 33, TR-9033-1, and is consistent in the use of the software domain and function identifications of that document. The document also extends and complements that information of TR-9033-1.

1. REVIEW OF EXISTING METRICS

The overriding theme for this survey and evaluation of applicable software metrics is to point up practical tools for acquisition and developmental program managers to make appropriate estimations and useful assessments on real-world problems. As Dr. J. Short of the Naval Underwater Systems Command indicated, "we get results from our metrics or we change them."

1.1 LITERATURE SURVEY

Priortizing the metrics literature reading of a project manager or V&V planner is a first step in managing a practical metrics program. The key concepts of quality and productivity factors and criteria within the project life cycle and framework provide the fundamentals for solid planning.

1.1.1 Framework Discussions

An understanding of the structure of the software metrics is facilitated by the definition of the software quality metrics framework. Though introduced by [RADC8502], the IEEE draft Standard for Software Quality Metrics has refined the framework.

The quality metrics framework provides an open-ended, hierarchy for organizing the conceptual elements of the quality metrics domain. These elements are quality attributes, criteria and measurements suitable for organizing quality control rooms and management tracking. The IEEE refinements include the concepts of "direct metrics" (for quality factors *like reliability*) and "subfactors" (e.g., for correctness). (Refer to Figure 1-1)

[IEEE (draft) P1061]
Standard for a Software Quality Metrics Methodology
Section 3 presents management-oriented objectives.
Section 4 presents the refined Software Quality Metrics
Framework.

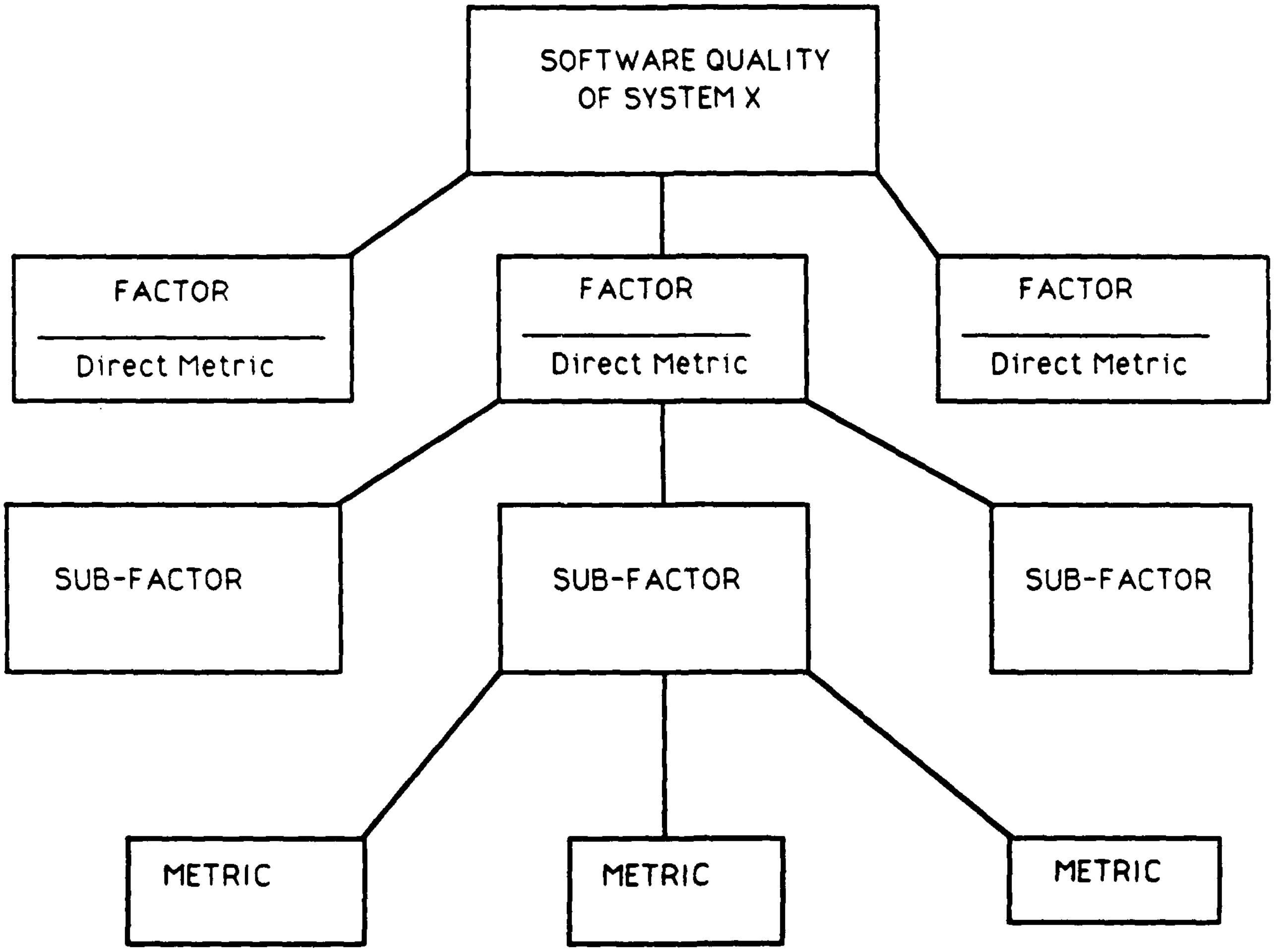
A good, comprehensive graduate text on software quality metrics was written by S.D. Conte, H.E. Dunsmore and V.Y. Shen in a collaborative effort between Purdue University and the Microelectronics and Computer Technology Corporation (MCC). The authors present many measurement and analysis examples.

[CONT86]
Software Engineering Metrics and Models, Benjamin Cummings, 1986.

Also, more particular to SDIO and the SDS, the (draft) IDA paper P-2132 (draft) [IDA8812] present a terser introduction that would be suitable for an SDS management pamphlet.

[IDA P-2132]
SDS Software Testing and Evaluation: A Review of the State-of-the-Art in Software Testing and Evaluation, Chapter 6. "Software Measurement Technology" present a current review of the state of the art of software quality metrics methodology.

REFINED SOFTWARE QUALITY METRICS FRAMEWORK



IEEE P1061 (DRAFT)

FIGURE 1-1

Chapter 7, "Software Reliability Assessment Technology" presents a concise, yet relevant discussion of the potential shortfall in current reliability metrics for the SDS software metrics requirements.

[RADC-TR-85-37, 3 vols]
Specification of Software Quality Attributes. Vol. 1 introduces the framework and concepts of qual'ty specification and evaluation. Volume 2 is a detailed guidebook for specification. Voiume 3 is the evaluation guide with sample checklists and worksheets.

[RADC-TR-87-171 v1]
Methodology for Software Reliability Prediction and Assessment, 3.1 "Software Quality Measurement Framework" includes the original framework discussion with definitions of all reliability concepts.

1.1.2 Management and Quality Indicators and Factors

The Air Force pamphlets AFSCP-800-14 and AFSCP-800-43 present a solid introduction to management (performance) and quality indicators. The pamphlets correlate the management and quality indicators, and then each quality indicator to its applicable software quality factors (as introduced by [RADC8502]. One disturbing statement "...there are no widespread tools available..." depends on the writers concept of "widespread", and is misleading -- tools are available.

The quality indicators pamphlets [-800-14] buries an important concept for management among two graphs and a chart:

4-13.... the degree of management insight can be significantly increased by using the software quality indicators. This combination of management and quality indicators should enable the contractor and the SPO to manage software development activities actively instead of reacting to software crises as they arise...

Note: The Army has replicated the Air Force pamphlets with few variations.

[AFSCP-800-14] Software Quality Indicators
[AMCP-70-14]

[AFSCP-800-43] Software Management Indicators
[AMCP-70-13]

Figure 1-2 presents the correlations of factors and indicators as per the pamphlets.

1.1.3 Management Methodology

As their draft evolves, Norman Schneidewind (USNPGS) and others on the IEEE Quality Metrics Standard Committee have emphasized the development of an extensible metrics management method for each software acquisition or evaluating organization. The approach taken by Dr. John Short and his metrics staff at NUSC is similar, emphasizing clear, well-understood metrics objectives, planning at the single project level with SOWs tailored to meet life cycle assessments points. The Air Force methodology guide emphasizes iterative planning, predicting and assessment. [IEEE(d)P1061], [RADC878A/B]

Software Management and Quality Indicators [AFSCP 800-14]

Quality	Management Indicators					
Indicators	CRU	SDM	RD&S	SPDT	C/SD	SDT
Completeness	X	X	X	X	X	X
Design Structure	X		X	X		X
Defect Density			X	X	X	X
Fault Density			X	X	X	X
Test Coverage			X	X	X	X
Test Sufficiency	X		X	X	X	X
Documentation	X	X	X			X

CRU = Computer resource utilization
SDM = Software development manpower
RD&S = Requirements definition & stability
SPDT= Software progress - development & test
C/SD = Cost/Schedule Deviations
SDT = Software development tools

Software Quality Indicators and Factors [AFSCP 800-14]

Quality	Software Quality Factors										
Indicators	Corr	Effc	Flex	Intg	Into	Main	Port	Reli	Reus	Test	Usab
Completeness	X					X		X		X	
Design Structure		X				X		X		X	
Defect Density	X					X		X		X	X
Fault Density	X					X		X		X	X
Test Coverage	X					X		X		X	
Test Sufficiency	X					X		X		X	
Documentation	X					X		X		X	X

Corr = Correctness
Effc = Efficiency
Flex = Flexibility
Intg = Integrity
Into = Interoperability

Main = Maintainability
Port = Portability
Reli = Reliability
Reus = Reusability
Test = Testability
Usab = Usability

Figure 1-2. Software Management /Quality Indicators & Factors

(Further discussion of Life Cycle importance is in Paragraph 1.6. The NUSC environment is discussed in Paragraph 3.1.)

1.1.4 Reliability

As Musa, Iannino and Okumoto introduce, "Reliability is probably the most important of the characteristics"...of software quality. They define reliability traditionally "the probability that the software will work without failure for a period of time to meet customer requirements." This subsumes many properties of software quality (correctness, friendliness, safety,...) but excludes modifiability, readibility (maintainability), which are less quantifiable.

[MUSA87]
Software Reliability - Measurement, Prediction, Application, McGraw Hill, 1987.

This comprehensive text provides practical measurement application guidance problems and solutions for managers and practicing software engineers, as well as the theoretical discussions needed for a college course. It uses the traditional hardware reliability approach taking advantage of the body of systems and control reliability knowledge. There is a reliance on probabilistic projections using random/stochastic techniques.

A key point about this approach: It is based upon unpredictability of programmer errors and unpredictability of execution conditions complicated by machine states. This does not account for the "known error" category (*non-fatal*, moderately reducing faults). Random implies "unpredictable" vs. "uniform". The use of techniques based upon this randomness assumption should not preclude other techniques which would presume some predictability about the failures, faults or defects.

Some further points made by John Musa in his articles should also be noted:

a. Reliability based on failure statistics is user oriented, more significant and suitable for setting (user-oriented) objectives (for prudent business services and typical information systems).

b. Defect/correction based quality measures are only significant in terms of contractor performance toward target goals.

Other key works on software reliability are from the Air Force RADC and the IEEE. The RADC work provides detail profiling of reliability and testing measurements. The more recent work of the IEEE Software Reliability Measurement Working Group is providing a refined dictionary and guidebook for reliability, including detail directory lookup for individual software metric parameters, a cohesive notation, instructions and evaluation references.

[RADC-TR-87-171]
Methodology (Vol. 1) and Guidebook (Vol. 2) for Software Reliability Prediction and Estimation

[IEEE 982.1]
(Draft) Standard Dictionary of Measures to Produce Reliable Software

[IEEE 982.2]
(Draft) Guide for the Use of the Standard Dictionary of Measures to Produce Reliable Software

As previously mentioned, the IEEE Quality Metrics Standard Committee is providing standard methodology with appendices of measurements, experience reports and validation guides [IEEE P1061].

1.1.5 Aggregate Indicators vs. Simple Measurements

The Air Force pamphlets, [CARD8707] and [GOIC89RC] indicate that primitive measures are to be aggregated with weighting factors based upon the relative value of the functions or modules effected when building the quality indicators. In the Waterfall SDLC these weights can be derived by requirements prioritization at the top level, and then functional factoring as the requirements are delineated. However, in a prototyping development, functional and performance goals may be used to establish relative system functional values.

Apart from the goal-oriented weights, the characteristics of the software architecture affect the relationship between software elements. John Musa has recently alerted us that the criticality of functions within the evolving system software architecture is a paramount consideration when considering software reliability, especially in regard to high-use servo functions or boundary control elements (e.g., Three Mile Island fault).

1.1.6 Validation and Refinement Articles

The technical literature is primarily concerned with the detail validations of particular reliability and maintainability assessment measures and more recently has been focusing upon the validity of predictions. There seems to be agreement among metrics experts that users should carefully plan and build understanding of the unique characteristics of their software and environment. Most of the software measures including balances or weights for development experience and people factors have shown to be more significant than unadjusted measures of the software alone.

We observed that the complexity and productivity models and tools do exist, but their interpretations and validations vary. Not much validation has been done on maintainability although tools do exist. Other issues such as software integrity, portability, usability, and reusability lack tools as well as validations. There are numerous validations and reliability models.

Some of the more significant results reached are listed below:

[TAKA8901] The results of this article indicate that models based on factors such as frequency of program specification change, programmer's skills, and volume of program design document are more reliable than conventional error prediction methods based only on program size.

[CARD8812] The results of this article indicate that complexity indicators based on structural (intermodule) complexity and local (intramodule) complexity seemed to be a useful tool for evaluating design quality before committing the design to code.

[LEW8811] The results of this article indicate that complexity measures are useful in quantifying the design and provide a guide to designing reliable software.

[BOEH8810] Barry Boehm and Philip Papaccio present some cost and quality observations specifically applicable to planned rapid deployment prototyping and other hybrid life cycle notions: low quality - low reliability software costs much more to maintain; high quality software tends to reduce its costs. High quality and low costs are promoted by personnel incentives, work environment and tool enhancements, and by software reuse with low rework. The authors cite use of their database of projects to validate the COCOMO model(s).

[*WEIS8810]* Weiss and Weyuker established an extended domain model of software reliability and used the Iannino, Musa, Okumuto rating criteria [IANN84]. The significance is that the notion of tolerable discrepancies of user service are introduced into a refined definition of software reliability, and the domain of program performance is enforced. This work was analytical and preliminary and needs followup, but promises applicability to large systems like the SDS, where the criticality of failures, the severity of defects, and the tolerance levels should be factored into reliability measures. The second author, Elaine J. Weyuker has independent software reliability work. Both were at NYU.

[DAVI8809] This short article analyzed the applicability of complexity measures (McCabe, Halstead) and lines of code in contrast to a group of new measures which analyze computer programs in terms of cognitive "chunks" as proposed by Lamergan, Mayer, Schneiderman and Solway (in various papers) "the software psychologists". The validating scope was limited to a set of small Fortran programs. Some significant observations are:

a. Halstead's length oriented "E" measure is more robust than anticipated.

b. Good debug time predictors also predicted the error counts well.

c. Constructive time for new programs depends more on size than complexity.

d. Maintenance time is strongly related to program complexity and lesser to overall size.

e. Data structure complexity is more significant to maintenance effort.

f. Control flow complexity is more significant to constructive effort.

g. Reinforces the view of Conte, Dunsmore and Shen that early SDLC phase metadata is fairly accurate at defining the actual data complexity developed. [CONTE8401]

[RAMA8808] This was a successful combination of Halstead's Software Science measures with McCabe's Cyclomatic Complexity Measure, using a set of weighting factors for the length and volume primitives. However, the combined results have no greater validity than the simple combination of separate measures.

[CARD8712] This article demonstrated that the basic relation of software science lacks empirical, as well as theoretic support. Future models of program construction process should be based more closely on observations of what programmers actually do.

[JEFF8707] This article found that the complex relationship between effort (productivity) and elapsed time variation cannot be measured by any of the current time-sensitive cost models.

[PERK8703] This article investigated a Navy-supplied Ada software and showed that an automatable, hierarchical, Ada-specific software metrics framework is an effective aid for improving the quality of Ada software.

[ABDE8609] This article examined 10 metrics: Jelinski and Moranda, Bayesian Jelinski-Moranda, Littlewood, Bayesian Littlewood, Littlewood and Verrall, Keiller and Littlewood, Weibull Order Statistics, Duane, Goel-Okumoto Model, and Littlewood NHPP model. The goal of the paper is to present an approach in deciding which is the most appropriate model to use. The results showed that there is no "best buy" among the 10 metrics. This is because predictive measures perform with varying degrees of accuracy on different software being studied. Therefore, the users need to be able to select and be sure that a chosen prediction metric is performing well for the type of prediction required.

[ALBR8411] This article demonstrates the equivalence between Albrecht's methodology to estimate the amount of the "function" or "function points" the software is to perform, and Halstead's "software science" model of "SLOC" measure. It also demonstrates the high degree of correlation between "function points" and the "SLOC" (source lines of code) of the program, and between "function points" and the work-effort required to develop the code.

[BASI8311A] This paper attempts to validate the Halstead's Software Sciences metrics, McCabe's Cyclomatic Complexity and other standard programs measures. The results of this article indicate that the Software Science metrics, Cyclomatic Complexity and other predictive metrics do not satisfactorily measure the errors occurred during development, and neither of these metrics really measure or predict effort or quality.

[HENR81] The article showed that the measurement of software quality for large-scale systems using information flow to represent the system interconnectivity is an important and viable technique.

1.1.7 Summarizing the Validation Literature

[LEW8811] and *[CARD8812]* agreed that complexity measures are useful in measuring and quantifying the design effort and provide a guide to designing reliable and high-quality software. In contrast, *[BASI8311A]* showed that the software science, cyclomatic complexity metrics and other predictive metrics do not satisfactorily measure the errors incurred during the development phase, and neither of these metrics really measures or predicts design effort or software quality. Ramamurthy and Melton *[RAMA8808]* examined software science and cyclomatic complexity and observed that combining these measures was as effective as making the separate assessments. They proposed a family of weighted measures which simultaneously detect the software characteristics that are detected by the software science measures and the cyclomatic number. *[GIBS8903]* studied the effect of system structure/complexity on system maintainability; specifically, what effect do structural differences have on maintenance performance, and are

structural differences measurable? The results indicate that the structural differences do impact performance and the metrics being validated (i.e., Halstead's E, McCabe's v(G), Woodward's K, Gaffney's Jumps, Chen's MIN and Benyon-Tinker's C2) can be used as project management tools.

From the article's results, some conclusions can be made:

a. There is an evolving consensus among complexity metrics validators that the application data used with the complexity models must be carefully selected and the conclusions must be scoped and scaled.

b. The validity metrics models for estimation and prediction requires continual updating.

c. There is a need to emphasized predictive measurands in the earlier phases of the life cycle (i.e., requirements and design).

d. The validation process must continue so that metrics can be effectively used to characterize and evaluate software products and processes.

e. A metrics methodology is needed for the effective use of metrics and each major system, development/acquisition must develop and tailor its own metrics program.

1.2 DATABASES

Technical interchanges and queries have been performed with the following information sources:

a. The TASQ Repository maintained for the U.S. Army AMCCOM Product Assurance Directorate in New Jersey. This database contains information on several thousand tools, characteristics and companies. The information is available via a series of in depth classification matrices and vendor descriptors. Appendix C contains selected portions of the TASQ database with potential direct or indirect support potential for the SDS. A complete TASQ expansion is separately bound in single copy as Appendix D (too physically large for reproduction).

b. The Air Force Systems Command Rome Air Development Center at Griffiss Air Force Base, New York. A number of technical reports from this source are referenced in the document list. The Data Analysis Center for Software (DACS) contains an extensive database and was extremely productive in obtaining information. (See references section)

c. The Institute of Electrical & Electronics Engineering (IEEE) contains an extensive set of standards, guidelines and draft standards. In addition, other technical IEEE publications are identified that served as an extensive source of information (e.g., Transactions on Software, Computer Magazine, Software Magazine, Spectrum Magazine, and others). The Association of Computing Machinery (ACM) provided additional metrics sources and related information; the "Communications of the ACM" publication served as one of the source documents. (see references section)

d. Portland State University's Metrics database and Dr. Wayne Harrison provided much assistance in acquiring information. (see references section)

e. George Mason University's Metrics repository and Dr. Ambrose Goicoechea provided direct assistance in acquiring metrics information. Professor Goicoechea was also instrumental in acquiring European surveys on the state-of-the-practice in metrics. (see references section)

f. The NASA Goddard Space Flight Center Software Engineering Laboratory and the University of Maryland have an extensive amount of metrics information. Meetings have been held with NASA's Frank McGarry, and University of Maryland (Dr. M. Zelkowitz). The Research Institute in Computer and Information Sciences (RICIS), University of Houston (Dr. C. McKay) was also contacted. RICIS is currently engaged in Mission and Safety Citical software initiatives with NASA's Johnson Space Center, Houston on the Space Station Software Support Environment (SSE). Software environment concerns for the SSE are similar to those that are expected for the SDI's (see references on Basili (Univ. of Md.), McKay)

g. The Naval Surface Weapons Center (NSWC) at Dahlgren, Virginia provided details about the Statistical Modeling and Estimation of Reliability Functions for Software (SMERFS) and summary data on Navy software support tools.

h. The Naval Underwater Systems Command (NUSC) provided descriptive data about their management methods for successful software metrics application.

i. The Headquarters Command at Kirtland AFB described their enhanced productivity model "REVIC" (outlined in Section 5).

1.3 CROSS REFERENCING MODELS BY ATTRIBUTES AND PARAMETERS

The following tables (1.3-1 through 1.3-6) summarize the metric models identified in the literature. Each table groups a set of models by its type of measurement employed. Currently the groups fall into six measurement categories. The categories are: 1) Time Between Failures, 2) Complexity, 3) Failure Count, 4) Fault Seeding, 5) Input Domain Based, and 6) Productivity. Each table indicates what specific software attributes are addressed by each model in the table. This identification establishes the framework for the mapping of software metrics to software types, processes, and domains or SDS subfunctions. Five of the six tables use the same attribute set; while the table on productivity uses an entirely different set. The models dealing with productivity, while concerned with performance and design attributes, are focused on project management issues such as cost, schedule, and the overall development process. For this effort, the software metric models addressing the nine attributes identified in will be covered. In addition to identifying metric models, this section also contains a table (1.3-7) listing the most commonly used parameters in the construction of software metrics.

The majority of tables use nine software attributes. These are the attributes identified in Subtask 1, TR-9033-1. Thirteen attributes (factors) were originally classified by Rome Air Development Center (RADC); however, based on familiarity with SDI component software requirements several attributes were combined to form the remaining set of eight. In addition, a ninth attribute was proposed in TR-9033-1. This attribute "throughput" addresses the user concerns of computational and communication throughputs. The following alignment shows the current nine attributes and the previously defined attributes that have been combined.

CURRENT ATTRIBUTE		OLD ATTRIBUTE
1.	Reliability	Reliability
2.	Survivability	Survivability
3.	Integrity	Integrity
4.	Efficiency	Efficiency
5.	Maintainability	Maintainability
		Correctness
		Verifiability
		Flexibility
		Expandability
6.	Usability	Usability
7.	Portability	Portability
8.	Reuse	Reusability
		Reuse
9.	Throughput	None

1.3.1 Software Quality Attribute Definitions

The following definitions are included here for completeness.

Reliability. The probability that software will not cause the failure of a system for a specified time under specified conditions.

Survivability. The built-in capability of the software to perform its required function when a portion of the system is inoperative.

Integrity. The degree to which the software controls unauthorized access to, or modification of, system software and data.

Efficiency. The degree to which the software performs its intended functions with minimum consumption of computer time and storage resources.

Throughput. The degree to which the software demonstrates its capability to process data identified as computational and/or communications related.

Maintainability. The effort required to locate and correct an error in the software.

Usability. The effort required to learn the human interface with the software, to prepare input, and to interpret output of the software.

Portability. The effort required to transfer the software from one hardware or software environment to another.

Table 1.3-1 Attributes for Time Between Failure Models

MODEL	ATTRIBUTES								
	Performance					Design			
	Reli	Surv	Intg	Effc	Thru	Usab	Main	Port	Reus
Time Between Failure									
Jelinski/Moranda	X	X							
Schick/Wolverton (Linear)	X	X							
Schick/Wolverton (Parabolic)	X	X							
Moranda (Geometric De-eut)	X								
Moranda (Hybrid Geomet Poiss)	X								
Goel/Okumoto	X								
Littlewood/Verrall	X								
Lloyd/Lipow	X								

Table 1.3-2 Attributes for Complexity Models

MODEL	ATTRIBUTES								
	Performance					Design			
	Reli	Surv	Intg	Effc	Thru	Usab	Main	Port	Reus
Complexity									
Halstead	X						X		
McCabe	X						X		
Woodward (Knot Counts)	X						X		
Chen (Nested Decision Stmts)	X						X		
Gaffney	X						X		
Benyon-Tinker	X						X		
Gilb's (Binary Decision)	X						X		
Chapin's Q	X						X		
Segment-Global Usage Pair	X						X		
Myer's (McCabe extension)	X						X		
Hansen's (McCabe/Halstead)	X						X		
Oviedo's (Data/Ctrl Flows)	X						X		

Effc = Efficiency
Intg = Integrity
Main = Maintainability
Port = Portability
Reli = Reliability

Reus = Reusability
Surv = Survivibility
Thru = Throughput
Usab = Usability

Table 1.3-3 Attributes for Failure Count Models

	ATTRIBUTES								
MODEL	Performance					Design			
	Reli	Surv	Intg	Effc	Thru	Usab	Main	Port	Reus
Failure Count									
Goel/Okumoto	X								
Schneidewind	X								
Goel	X								
Musa	X								
Shooman	X								
Moranda	X	%							
Jelinski/Moranda	X	%							
Moranda	X	%							
Schick/Wolverton	X	%							
Goel/Okumoto (Gen Poisson)	X	%							
Brooks/Motley	X	%							
IBM Poisson	X	%							

% Each model contains a different representation for a hazard function that may be adapable to support survivability

Table 1.3-4 Attributes for Fault Seeding Models

	ATTRIBUTES								
MODEL	Performance					Design			
	Reli	Surv	Intg	Effc	Thru	Usab	Main	Port	Reus
Fault Seeding									
Mills	X								

Effc = Efficiency
Intg = Integrity
Main = Maintainability
Port = Portability
Reli = Reliability

Reus = Reusability
Surv = Survivibility
Thru = Throughput
Usab = Usability

Table 1.3-5 Attributes for Input Domain Based Models

MODEL	ATTRIBUTES								
	Performance					Design			
	Reli	Surv	Intg	Effc	Thru	Usab	Main	Port	Reus
Input Domain Based									
Nelson	X								
Ho	X								
Ramamoorthy/Bastani	X								

Effc = Efficiency
Intg = Integrity
Main = Maintainability
Port = Portability
Reli = Reliability
Reus = Reusability
Surv = Survivibility
Thru = Throughput
Usab = Usability

Table 1.3-6 Attributes for Productivity Models

MODEL	ATTRIBUTES			
Productivity	Size	Cost	Comp	Devp
Software Size	X			
Personnel		X		
Volatility		X	X	
Resource Utilization				
Complexity	X	X	X	
Schedule Progress	X	X	X	X
Design Progress			X	
CSU Development Progress	X		X	X
Testing Progress		X	X	X
Incremental Release Content	X	X	X	X

Size = Size of the Program
Cost = Total Cosyt
Comp = Completion Date
Devp = Effect of the Development Process

Table 1.3-7 Parameters Associated with Metric Classes

	METRIC CLASSES				
	T	C	F	F	I
PARAMETERS	B	P	C	S	D
	F	X			B
Observed time between failures	X				
Calculated hazard function	X				
Subjective Random Variable	X				
Cum No. of Observed failures			X	X	
No. of faults detected during a time period			X	X	
No. of faults seeded into program				X	
Failure rate			X	X	
Expected No. of faults to be detected			X	X	
Amount of execution time			X		
No. of control transfers		X	X		
No. of instructions in the program		X	X		
Debugging time since time of integration			X		
No. of faults corrected			X		
No. of entries/exits per module		X			
Software science measures		X			
Design structure		X			
Data flow complexity		X			
Requirements traceability		X			
Software documentation		X			
No. of operands & operators		X			
"Binary" decisions in program logic		X			
No. of external interfaces		X			
Segment use of global variables		X			
No. of intersecting control statements		X			
No. of compound predicates		X			
Inputs by subdomains					X

Metrics Classes:
TBF = Time Between Failures
CPX = Complexity
FC = Failure Count
FS = Fault Seeding
IDB = Input Domain Based

Reusability. The degree to which the software can be used in multiple applications.

1.4 SCREENING INAPPROPRIATE SOFTWARE METRICS

A review of the metrics contained in paragraph 1.3 reveals that no single metric is capable of supporting decisions across an entire software domain as defined in Subtask 1, TR-9033-1. Each metric supports measurements in one or two software attributes; e.g., Reliability, Maintainability, etc. The implication here, is that perhaps several metrics will need to be combined to form a composite model capable of supporting decisions on the 36 SDS subfunctions or the defined software domains. The analysis concludes that no available metric should be dropped from consideration.

1.5 MAPPING SOFTWARE METRICS TO TYPES AND PROCESSES

Each of the software metrics classes presented is mapped into its applicable major function and subfunction, using the attribute rankings developed in TR-9033-1.

1.5.1 Ranking Attributes by Software/Process Types

First, the characteristics of each subfunction were studied to determine the relative weights of the attributes. This analysis was performed in Subtask 1 and is explained in SDIO Task 33, TR-9033-1 SDS Software Measurement Requirements, Para. 2.3 "Characteristics of Software." The following characteristics were postulated: a) Criticality, b) Embedded vs. general purpose, c) Space/Ground based, d) Life cycle, e) Algorithmic Content, f) Size, g) Risk, and H) Intended use.

Then, in TR-9033-1, Para. 2.4, the characteristics of each subfunction were used to determine the attribute rankings for each SDS Subfunction.

Table 1.5-1 summarizes the subfunction attribute rankings, concentrating upon the "high" and "medium" rankings; the remaining subfunction-attribute correlations were all "low."

1.5.2 Subfunction Metrics Applicabilities

The applicability of the metrics classes and models to each of the quality attributes was presented in Para. 1.3 and Tables 1.3-1 through 1.3-6. These applicabilities are used as a basis to derive the subfunction metrics applicabilities.

Table 1.5-2 presents the extrapolations to metrics classes from the attribute.and applicability rankings, and Table 1.5-3 presents the derivation rules used.

1.5.3 Development Process Applicability

Prior to the analytical evaluation of the metrics in regard to the subfunction requirements, (presented in Para. 2) we need to consider how the development process can affect metrics applicabilities. The possibility of employing off-the-shelf and preused software componenets presents a different software metrics environment than the single-vendor prototype concept, and the traditional requirements-based, multi-phased development.

Table 1.5-1. Software Functions with Ranked Attributes (from TR-9033-1)

Major Function	Subfunction	High/Med Applicable Attributes								
No. Title	No. Title	Reli	Surv	Intg	Thru	Usab	Main	Port	Reus	Effc
1 Detect										
	1 Plumes	H	H	H	H					M
	2 Coldbodies	H	H	H	H					H
	3 RF	H	H	H	H		M	M		M
2 Identify										
	4 Resolve Obj.	H	H	H	M					M
	5 Discr	H	H	H	M					M
	6 Assess Kills	H	H	H	M			M		M
3 Track										
	7 Correlate	H	H	H	H					M
	8 Initiate	H	H	H	H					M
	9 Estimate	H	H	H	H			M		M
	10 Predict I&I	H	H	H	M					M
4 Communicate										
	11 Interplatform	H	H	H	H				M	H
	12 Ground-Space	H	H	H	M		M	M		M
	13 Ground	M	M	M	M	M	H	M	H	
5 Assess										
	14 Threat	H	H	H						
	15 SDS	H	H	H						
6 Wpn Contrl										
	16 SBI Asgn&Ctrl	H	H	H	H					M
	17 GBI Asgn&Ctrl	H	H	H	M		H			M
	18 SBI Guide&Ctrl	H	H	H	H					H
	19 GBI Guide&Ctrl	H	H	H	M		H			M
7 Platfm Mgt										
	20 Cmd Env Ctrl	H	M	H	M	M	H	M		M
	21 Ctrl Onbd Env	H	H	H	H					M
	22 Cmd Attit&Pos	H	M	H	M	M	H	M		M
	23 Ctrl Attit&Pos	H	H	H	H					M
	24 Sense Status	H	H	H	M					M
	25 Assess Status	H	M	H	M	M	H	M		M
	26 Cmd Reconfig	H	M	H	M	M	H	M		M
	27 Reconfigure	H	H	H	H					M
8 Spprt Dev										
	28 Tools	M	M	M		M	M	M		M
9 Simulate										
	29 HWIL	M	H	H	M	M	M			M
	30 Demonstration	M	H	M	M	M	M	M		M
10 Spprt Acqu										
	31 Developer Test	M	M	M	M	H	H	H	H	M
	32 Developer Env	H	M	H	M	H	M			M
	33 Factory Test	M	M	M		H	H	H	H	
	34 Acceptance Test	M	M	M	M		M			M
11 Spprt Mgt										
	35 MIS DB Maint	M	M	H	M	H	H	H	H	M
	36 Track Mgt Info	M		H		H	H	H	H	M

ATTRIBUTES: Reli = Reliability
Surv = Survivability
Intg = Integrity
Effc = Efficiency
Port = Portability
Main = Maintainability
Usab = Usability
Reus = Reusability
Thru = Throughput

Table 1.5-2 Deriving Metrics Rankings from Subfunction Attributes

FUNCTIONS	ATTRIBUTES									METRICS CLASSES				
No. Subfunction	Reli	Surv	Intg	Thru	Usal	Mair	Port	Reu:	Effc	TBF	Cpx	FC	FS	IDB
Detect														
1 Plumes	H	H	H	H					M	H	H	H	H	H
2 Coldbodies	H	H	H	H					H	H	H	H	H	H
3 RF	H	H	H	H		M	M		M	H	H	H	H	H
Identify														
4 Resolve Obj.	H	H	H	M					M	H	H	H	H	H
5 Discriminate	H	H	H	M					M	H	H	H	H	H
6 Assess Kills	H	H	H	M	M	M	M		M	H	H	H	H	H
Track														
7 Correlate	H	H	H	H					M	H	H	H	H	H
8 Initiate	H	H	H	H					M	H	H	H	H	H
9 Estimate	H	H	H	H			M		M	H	H	H	H	H
10 Predict I&I	H	H	H	M					M	H	H	H	H	H
Communicate														
11 Interplatform	H	H	H	H				M	H	H	H	H	H	H
12 Ground-Space	H	H	H	M		M	M		M	H	H	H	H	H
13 Ground	M	M	M		M	H	M	H		M	M	M	M	M
Assess														
14 Threat	H	H	H							H	H	H	H	H
15 SDS	H	H	H							H	H	H	H	H
Wpn Control														
16 SBI Asgn&Ctrl	H	H	H	H					M	H	H	H	H	H
17 GBI Asgn&Ctrl	H	H	H	M		H			M	H	H	H	H	H
18 SBI Guide&Ctrl	H	H	H	H					H	H	H	H	H	H
19 GBI Guide&Ctrl	H	H	H	M		H			M	H	H	H	H	H
Platfm Mgt														
20 Cmd Env Ctrl	H	M	H	M	M	H	M		M	H	H	H	H	H
21 Ctrl Onbd Env	H	H	H	H					M	H	H	H	H	H
22 Cmd Attit&Pos	H	M	H	M	M	H	M		M	H	H	H	H	M
23 Ctrl Attit&Pos	H	H	H	H					M	H	H	H	H	H
24 Sense Status	H	H	H	M					M	H	H	H	H	M
25 Assess Status	H	M	H	M	M	H	M		M	H	H	H	H	M
26 Cmd Reconfig	H	M	H	M	M	H	M		M	H	H	H	H	M
27 Reconfigure	H	H	H	H					M	H	H	H	H	H
Spprt Dev														
28 Tools	M	M	M		M	M	M	M	M	M	M	M	M	
Simulate														
29 HWIL	M	H	H	M	M	M			M	H	M	H	M	M
30 Demonstration	M	H	M	M	M	M	M		M	H	M	H	M	M
Spprt Acqu														
31 Developer Test	M	M	M	M	H	H	H	H	M	M	H	M	M	M
32 Developer Env	H	M	H	M	H	M			M	H	H	H	H	H
33 Factory Test	M	M	M		H	H	H	H		M	H	M	M	M
34 Acceptance Test	M	M	M	M		M			M	M	M	M	M	M
Spprt Mgt														
35 MIS DB Maint	M	M	H	M	H	H	H	H	M	M	H	M	M	M
36 Track Mgt Info	M		H		H	H	H	H	M	M	H	M	M	M

METRIC CLASSES: "TBF" -- Time Between Failures
"FC" -- Failure Count Models "Cpx" -- Complexity
"IDB" -- Input Domain Based "FS" -- Fault Seeding

Table 1.5-3: Metrics Ranking Derivation Rules

ATTRIBUTE RANKING		METRIC CLASS/MODEL RANKING
<RELIABILITY "RELI">	==>	<TIME BETWEEN FAILURES "TBF"> &<COMPLEXITY "CPX"> &<FAILURE COUNT "FC"> &<FAULT SEEDING "FS"> &<INPUT DOMAIN BASED "IDB">
<SURVIVABILITY "SURV">	==>	<TBF> & <FC>
HIGH "H"	==>	H
MEDIUM "M"	==>	M
H & M	==>	H
H & H	==>	H
M & M	==>	M

Two classes of metrics assume access to source code: Complexity Models (Halstead, McCabe, Woodward, etc.) and the Mills Fault-Seeding Model, which also requires the alteration of the source or the software adaptation data (parameters, etc.). The failure-sensitive (TBF & FC) and input domain (IDB) and many productivity metrics models are independent of the source code, and do appear applicable to hybrid development processes.

Other developmental process implications are discussed next in regard to the software development life cycle (SDLC).

1.6 LIFE CYCLE

1.6.1 Life Cycle Models

This section on life cycles has a fourfold purpose:

a. To establish a reference life cycle to support the categorization of metrics, and the particular phase with which they are associated or used;

b. To establish a foundation and reference point from which subsequent and different life cycle forms can be derived or related;

c. To establish a reference model that can be used in the identification of new metric forms and types, and the phase in which they will be employed;

d. To provide the basis for establishing new metrics methodologies and usage emphasis.

To support the categorization of metrics, a reference or standard life cycle required identification together with its identified phases. The software life cycle used is that defined in [MIL-STD-2167A]. A number of other life cycle variations were found (RADC878A, IEEE Std. P982.2/D6, [BOEH81]) and examined. Life cycles were found to vary in the number and definition of phases. However, these differences for the most part were found to be organizational, with the content essentially the same as Tables 1.6-1,2&3 illustrate. Life cycle variations were minor, and all of these models can be mapped into each other with relative ease, as well as the waterfall model of TR-9033-1.

Newer life cycle models (i.e., Spiral, technology, iterative) that more appropriately support complex development activities such as prototyping and reusability paradigms were not considered for evaluation. The newer models have not been used to the extent that the classical waterfall model has, and thus not much experience and data are available. However, their iterative nature, ("build a little, test a little, correct/adjust, repeat the process again"), coupled with the formality of some of the prototypes used in this iterative process, is more appropriately suited to modem day developments. Large and complex system developments that must endure changing requirements, as well as undergo evolutionary or incremental changes must be supported by iterative processes that provide the ability to revisit and reexamine design incursions and baseline changes. It is, therefore, recommended that a more indepth examination of iterative life cycle models be made in the context of this report.

Items a. and d. are considered within the scope of this task, items b. and c. are outside the scope of this effort. The basis for the scope statements is the pragmatic task emphasis of identifying existing and available software and system measurement, and variably reliable predictors of high utilitarian value to managers and technical personnel.

A few observations about the referenced software development life cycles contained in Tables 1.6-1,2,3:

- User and developer bidirectional communications and information flow are not well supported (e.g., human engineering and software engineering). This is indicative of the lack of feedback paths within the life cycles required to resolve deficiencies, issues and problems as they arise;

- Events and activities are sequential in nature, with very little support for iterative processes. The 2167A life cycles support prototyping and design reusability. Iterative processes are required to tune or refine prototypes as design information evolves and is elaborated upon;

- Resulting products from such are limited and constrained. The varying degrees of prototype and reuse formalisms will require tailoring of existing specifications, as well as require new ones. Additionally, associated design reviews will also require customizing (e.g., reuse preliminary design review, prototype specification design review).

More sophisticated configuration, delivery and requirements packaging, such as incremental development, incremental deployment, evolutionary development, transactional threading, and technology insertion have given rise to newer, less well defined models. Models to support various advanced life cycle requirements have had slow acceptance (formal prototype, executable prototype, spiral, automation-based model).

The applicability of each metric to the life cycle phases must be clearly identified. This is essential in planning for effective visability and predictability in advance of development. A metric used to assess code structure is one that is invoked late in the development life cycle where much time and resources have already been consumed (post-facto). A metric or predictor used in the early or initial life cycle phases (i.e., requirements/design synthesis) can provide anticipatory or predictive insight before long term resources are committed, where it is much more cost-effective [BOEH81] to make design tradeoffs and correct problems.

TABLE 1.6-1

COMPARING MIL-STD-2167A & RADC-TR-87-171

MIL-STD-2167A		RADC-TR-87-171	
No.	Phase	No.	Phase
1	S/W Reqts. Analysis	1	S/W Reqts. Analysis
2	Preliminary Design	2	Preliminary & Detailed Design
3	Detailed Design		
4	Coding & Unit Test	3	Coding & Unit Test
5	CSC Integration & Test	4	CSC Integration & Test
6	CSCI Testing	5	CSCI Testing
7	System Integ. & Testing	6	System Integ. & Testing
8	Oper. Test & Evaluation	7	Oper. Test & Evaluation
9	Deployment & Distribution	8	Production & Deployment

TABLE 1.6-2

COMPARING MIL-STD-2167A & [BOEH81] WATERFALL PHASES

MIL-STD-2167A		[BOEH81] WATERFALL LIFE CYCLE	
No.	Phase	No.	Phase
1	S/W Reqts. Analysis	1	Software Reqts./Validation
2	Preliminary Design	2	Product Design/Verification
3	Detailed Design	3	Detailed Design/Verification
4	Coding & Unit Test	4	Code/Unit Test
5	CSC Integration & Test	5	Integ./Product Verification
6	CSCI Testing		
7	System Integ. & Testing	6	Implementation System Test
8	Oper. Test & Evaluation	7	Op's & Maint./Revalidation
9	Deployment & Distribution		

TABLE 1.6-3

COMPARING RADC-TR-87-171 & [BOEH81] WATERFALL PHASES

RADC-87 LIFE CYCLE		[BOEH81] WATERFALL LIFE CYCLE	
No.	Phase	No.	Phase
1	S/W Reqts. Analysis	1	Software Reqts./Validation
2	Preliminary and Detailed Design	2	Product Design/Verification
		3	Detailed Design/Verification
3	Coding & Unit Test	4	Code/Unit Test
4	CSC Integration & Test	5	Integ./Product Verification
5	CSCI Testing		
6	System Integ. & Testing	6	Implementation System Test
7	Oper. Test & Evaluation	7	Op's & Maint./Revalidation
8	Production & Deployment		

As indicated in Subtask 1, TR-9033-1, approximately 70% of all "software errors" actually occur early in the development cycle. This gives great importance to identifying and implementing those metrics which can be applied in the early design phases. An examination of a well defined characteristic life cycle model (e.g., RADC's) as presented in Table 1.6-4, with its associated reliability measure model (Table 1.6-5) reveals the following:

- The identification of a prototyping phase and an extensible reusability dependent life cycle (i.e., reusable architecture, design, specifications and code) requires the identification of new metrics or redefinition of others. Thus, the entry for reuse metrics (Table 1.6-4) would also appear in the preliminary and detailed design phase; while a new one would be entered in this same phase for prototyping.

- Secondly, the associated reliability measurement model of Table 1.6-5 would require similar adjustments in its metrics formula representation.

- Thirdly, a shift in emphasis would occur, from estimation to predictive metrics, if emphasis were placed on prototyping a system in order to obtain early visibility into the design and ferret out initial design issues.

The impact on the reliability model predictive and estimation metrics (Table 1.6-5) relationships (see reference for formuli details) can be altered significantly if new terms, such as NR and NP are added to this established model.

Furthermore, in both the RADC and IEEE models, predictive metrics (category 1) used in the early life cycle phases are the ones where the least amount of information is available and where formal relationships are least understood.

RADC LIFE CYCLE

	METRICS	Concept Devel'pt Acquis'n Initiat'n	Mission System Softw. Defin'tn	System Requ'ts Anal.	Prelim. and Detail Design	Coding and Units Testing	CSC Integr'n and Test	CSCI Test	System Integr'n and Test	Oper. Test and Eval.	Produce & Deploy
PREDICTIVE	* Application Type (A)	X	X								
	* Development Environment (D)			X							
	* Software Characteristics (S)										
	+ Anomaly Mgt (SA)				X						X
	+ Traceability (ST)				X	X					X
	+ Quality Review (SQ)				X						X
	- Language Type (SL)					X					X
	- Program Size (SS)					X					X
	- Modularity (SM)					X					X
	- Extent of Reuse (SU)					X					X
	- Complexity (SX)					X					X
	- Standards Review (SR)				NR, NP	X					X
ESTIMATION	Failure Rate During Test (F)						X	X	X	X	
	+ Test Effort (TE)						X	X	X	X	
	+ Test Methodology (TM)						X	X	X	X	
	+ Test Coverage (TC)						X	X	X	X	
	- Workload (EW)						X	X	X	X	X
	- Input Variability (EV)						X	X	X	X	X

New Metrics Required:
NR--- For Extensive Reuse
NP -- For Prototyping

Table 1.6-4: Composite Metrics Across the Waterfall Life Cycle

TABLE 1.6-5

SOFTWARE RELIABILITY MEASUREMENT MODEL
RADC-TR-87-171 PREDICTIVE AND ESTIMATION METRICS

PART 1: PREDICTIVE METRICS

Application Type	A		
Development Environment	D		
Software Characteristics	S		
Requirements and Design Representation		S1	
Anomaly Management			SA
Traceability			ST
Quality Review Results			SQ
Software Implementation		S2	
Language Type			SL
Program Size			SS
Modularity			SM
Extent of Reuse			SU
Complexity			SX
Standards Review Results		SR	

Rp = A x D x S where

S = S1 x S2
S1 = SA x ST x SQ
S2 = SL x SS x SM x SU x SX x SR

PART 2: ESTIMATION METRICS

Failure Rate During Testing	F	
Test Environment	T	
Test Effort		TE
Test Methodology		TM
Test Coverage		TC
Operating Environment	E	
Workload		EW
Input Variability		EV

RE = F x T, during testing where

T = TE x TM x TC and

RE = F x E, during OT&E where

E = EW x EV

Focusing on MIL-STD-2167A thus establishes the need for the definition of a more extensive and formal life cycle that can address the activities and processes for prototyping and reusability among others.

1.6.2 Life Cycle Experience Classificatioı

Table 1.6-6 represents an experience classification matrix of measures and the life cycle phase in which their use is appropriate. The table divides the measures into three categories:

Category 1 - identifies measures that have been formalized (e.g., by a mathematical foundation)
and have limited or insufficient operational validation

Category 2 - identifies measures that have been formalized and have a moderate experience validation

Category 3 - identifies measures that have been formalized and have an extensive experience basis.

The table serves to identify measures (category 1) that may have a higher utilitarian value than the currently utilized metrics of category 3 [IEEE P982] that are well understood and industry recognized (e.g., McCabe, Halstead). It is interesting to note that category 3 measures fall into the later phases of the life cycle further supporting the claim that they are post-facto measures (i.e., occur from the coding phase to the operations and maintenance phase) for the most part where significant resources have been consumed and committed. While category 3 measures represent approximately 20% of the total and are represented by the majority of tools that exist in the marketplace, category 1 measures account for approximately 50% and are not supported by many tools and environments. The remainder fall into category 2 (approximately 30%).

A contributing factor to the unavailability of metrics information rigor in the early phases of the life cycle is the fact that use of formal notation to support requirements and design synthesis is not available or utilized within the industry for the most part. The sparse use of formal notation within the life cycle reveals a serious technology and communications gap between the systems and software engineering communities. Use of a formal syntax (e.g., BNF notation) or system design language, with the same level of formalism as those that exist in the implementation phase of the life cycle (i.e., programming language) has the benefit of being machine processable and analyzable.

The coexistence of both a formal system design and a programming language provides the basis for applying complexity, efficiency, reliability, etc., measures in a more consistent and predictive manner. The existence of a formal system design language allows an early assessment of design integrity and completeness that can in turn be used to predict code and structural complexity, and ease of integration. Formal notation would provide metrics measurements and models with a more extensive and effective range of application.

Part 1: LIGHT EXPERIENCE METRICS BY LIFE CYCLE PHASE

Category 1 Metric	Conc	Rqmts	Des	Implem	Test	Intg/Tes	O&M
Cum. Failure Profile	x	x	x	x	x	x	x
Cum. Failure Profile	x	x	x	x	x	x	x
Functional/ Modular Test Coverage					x	x	x
Defect Indices	x	x	x	x	x	x	x
Errors Distributions		x	x	x	x	x	x
S/W Maturity Index	x	x	x			x	x
Number Entry/Exits Per Module			x	x			x
Graph-Theoretic Cmplexity for Arch.		x	x				x
Design Structure			x				x
Software Purity Level					x	x	x
Requirements Complicance		x					x
Data or Info. Flow Complexity			x	x			x
Residual Faults Count					x	x	x
Testing Sufficiency					x	x	
Required Software Reliability	x	x	x	x	x	x	x
Software Release Readiness					x	x	
Test Accuracy					x		
Indep. Process Reliability						x	x
Combined HW/SW Operational Reli.					x	x	x

Part 2: MODERATE EXPERIENCE METRICS BY LIFE CYCLE PHASE

Category 2 Metric	Conc	Rqmts	Des	Implem	Test	Intg/Tes	O&M
Fault Density	x	x	x	x	x	x	x
Cause & Effect Graphing		x	x	x	x		x
Manhours per Major Defect Detected		x	x	x	x	x	x
Number of Conflicting Rqmts		x					x
Minimal Unit Test Case Determination				x	x		x
Run Reliability					x	x	x
Estimated Number of Faults Remaining					x	x	x
Test Coverage		x	x		x	x	x
Reliability Growth Function					x	x	x
Software Documentation & Source Listing				x	x	x	x
Completeness		x	x				x
System Performance Reliability		x	x	x	x	x	x

Part 3: GREATER EXPERIENCE METRICS BY LIFE CYCLE PHASE

Category 3 Metric	Conc	Rqmts	Des	Implem	Test	Int/Tes	O&M
Defect Density	x	x	x	x	x	x	x
Requirements Traceability		x	x				x
Software Science Measures				x			x
Cyclomatic Complexity				x	x		x
Mean Time To Discover The Next K Faults					x	x	x
Failure Analysis Using Elapsed Time					x	x	x
Mean Time To Failure					x	x	x
Failure Rate					x	x	x

Table 1.6-6: SOFTWARE METRICS EXPERIENCE CLASSIFICATION

1.6.3 Summary

Paragraphs 1.6.1 and 1.6.2, together with the other sections of this report, are intended to provide the basis for the following:

a. Much more metric information and formalism is concentrated on the later phases of the life cycle (i.e., from coding/implementation to the operations and maintenance phases). The reference section provides additional strong support for the statement.

b. The cost to correct errors or deficiencies in the later phases increases by at least one to two orders of magnitude. This is attributable to the consumption of resources and labor intensive activities that have occurred by the time that implementation or coding is reached [BOEH81].

c. The coding and related phases (e.g., CSC phase) of the life cycle are the least significant cost drivers. It is recognized, thus specific references are not required, that maintenance is the costliest phase, as a result of redesign, requirements changes, poor design visibility and specifications.

d. Current development efforts that have focused on the early phases of the life cycle, predictive metrics and the early detection of problems have had very good productivity (cost and schedule) and have produced quality software with fewer catastrophic errors. IBM's Space Shuttle Program, NUSC's Submarine Combat System, Unisys Trident Submarine Navigation Program, and Teledyne Brown's SDI SIE and N-SITE programs are examples of such. Though these efforts are few in number, a consistent theme is emerging.

Indications are that focusing on predictive/early life cycle use of metrics and focusing on the early life cycle phases themselves, will provide a better quality and cost-effective design. Yet predictive metrics and the measures of category 1 (Table 1.6-6) are the ones with the least experience factors and understanding.

2. ANALYTIC EVALUATION OF SOFTWARE METRICS

2.1 FEATURE ANALYSIS

As outlined in Subtask 1, TR-9033-1, several software domains can be constructed using combinations of characteristics and factors applied to 36 SDS subfunctions. An assessment of how each applicable software metric can be applied within each SDS subfunction will be presented. Each table in Appendix B identifies the SDS subfunction and the nine software attributes along with their relative rankings as presented in Subtask 1, TR-9033-1, for that subfunction. Each relative ranking is assigned a score. The scores are arbitrarily assigned a value from 1, for low, to 5, for high.

A vector is created, for each identified metric, using the scores assigned from the relative rankings given in TR-9033-1. For example, the first subfunction identified is "Detect Plumes". Associated with that subfunction are the following attributes with their relative rankings. Beside each relative ranking is its associated score.

Quality Attributes	Relative Rankings	Score
Reliability	High	5
Survivability	High	5
Integrity	High	5
Efficiency	Moderate	3
Throughput	High	5
Usability	Low	1
Maintainability	Low	1
Portability	Low	1
Reuse	Low	2

Taking the McCabe metric from Table 1.3-2, note that this metric is used in measuring reliability and maintainability. For Detecting Plumes, the need for reliability as defined in TR-9033-1 is high, while the need for maintainability is low. Since these are the only two attributes measured by this metric the following vector is created:

Detect Plumes
TR-9033-1 Rankings

	H	H	H	H	L	L	L	L	M
	Reli	Surv	Intg	Thru	Usab	Main	Port	Reus	Effc
McCabe	5	0	0	0	0	1	0	0	0

Recognize that each subfunction has a different set of attribute rankings and therefore creates a different vector set.

Using the McCabe metric for another SDS subfunction, e.g., "Command Environment Control", the vector is different and it appears more relevant to this subfunction than Detecting Plumes.

Command Environment Control

TR-9033-1 Rankings

	H	M	H	M	M	H	M	L	M
	Reli	Surv	Intg	Thru	Usab	Main	Port	Reus	Effc
McCabe	5	0	0	0	0	5	0	0	0

The McCabe metric, though still deficient in six of the eight attributes, matches the requirements of the Command Environment Control subfunction better since it would be able to measure 2 of the 3 attributes designated as a high need.

The reader should note the large number of zeroes included in the vectors shown in Appendix B. This indicates that the established metrics do not support many of the SDS required attributes. It appears that much work still needs to be accomplished in creating metrics that address those attributes. One method of attacking this deficiency would be to use an emerging class of metrics designated as multi-attribute metrics. Some pioneering work is currently on-going at George Mason University and other academic institutions around the country. In fact, the National Science Foundation sponsors a Special Interest Group on Multi-Criteria Decision Making (MCDM) which meets periodically to discuss the state-of-the-art in software measurement.

An alternative approach would be to investigate the feasibility of creating metrics for those attributes not currently addressed. After those metrics are created, a composite (or collective) set of metrics could be applied to each SDS subfunction. The greatest payback would be achieved by applying those composite or multi-attribute metrics that address the attributes with high or moderate relative ranking. If funds still remained in the quality assurance purse after covering higher level attributes, then the attributes designated as low ranking could be addressed.

Caution must be exercised in interpreting the values (scores) in the created vectors. No one should make a judgement that one metric is "better" than another because a vector value is larger, e.g., 5 vs. 1. Remember the scores are highly dependent on the relative ranking derived in TR-9033-1 for each SDS subfunction. Even the same metric can have several different vectors, depending again on the SDS subfunction being evaluated. The assessment that can be made for a vector with higher scores is that the application of that metric to its specific SDS subfunction will provide information about an attribute assessed, in TR-9033-1, to be more critical. This report does not claim to rank the metrics. It attempts to show where specific metrics can best be applied effectively across all 36 SDS subfunctions.

The well-defined metrics which were analyzed in this section are generally applied late in the software development life cycle (SDLC). They are normally classified as estimation type metrics rather than predictive. The current trend is to develop and apply predictive metrics early in the SDLC to aid not only program managers, but also development personnel in identifying unfavorable trends early in development where corrective action can be taken with minimum cost and schedule impacts. As previously presented in Table 1.6-4, the more research and development done to inspire predictive metrics for the preliminary and detailed design phase of the life cycle, the better the control will be on future developmental efforts.

3. EXPERIENCE ASSESSMENT

A number of major programs were identified as potential sources that would provide field experience and relevant insight into software metrics. Several programs were eliminated initially due to their states of maturity or information availability. NASA's Space Station initiatives are in their initial phases, thus a shift to other programs with a greater experiential base was merited (e.g., NASA/IBM Shuttle program). However, that is not to say that the Space Station program does not have a quality or metrics program. Some of the RICIS literature and effort clearly focuses on mission and safety critical software and related issues for the Space Station.

The Joint Tactical Fusion program has a viable metrics initiative. Though information requested is forthcoming, the classified nature of the program precludes the screening of relevant metrics information in time for this report. However, review of models used and metrics initiatives are consistent with and support conclusions reached in this report.

Formally instituted metrics programs on major systems acquisition are not readily identifiable, with the exception of those noted in the following sections. For the most part, metrics efforts are contained within larger quality assurance programs or as smaller independent research initiatives within DoD. However, where metrics programs have been highlighted as part of major program acquisitions, much success has followed those developments/acquisitions.

3.1 THE NAVAL UNDERWATER SYSTEMS COMMAND (NUSC)

The NUSC facility at Newport, Rhode Island, has been using a suite of software metrics to manage submarine combat systems acquisitions for the last four years. Dr. John Short, Dept. Manager, was instrumental in setting the management guidance for a cohesive method of contractor oversight using software metrics throughout the life cycles of each of the emerging developments.

The management concept at work is risk management: to manage project and resource risks by assessments, predictions, validations and risk mitigations. The process starts with RFP development and proceeds apace of the life cycles. Key players within NUSC and the contractors are given training and briefings by the NUSC software metrics staff.

The NUSC database and experience is focused on the Navy Submarine Combat System Applied Software Metrics Program. NUSC has approximately 10 million lines of source code (MSLOC) in process with projects varying between 0.5 and 4.0 MSLOC.

The NUSC software metrics program applications have initially centered on CMS-2 language applications, with two recent exceptions, the CCS MK 2 and AN/BSY-2 programs using Ada and MIL-STD-2167. Key tools used in the NUSC program are LOTUS 1-2-3 as the spreadsheet for information collection of such items as faults, problem trouble reports (PTR), and PTR testing, supported by an ORACLE database; at least 4 development models (SLIM, COCOMO, SECOMO, SASET); 2 reliability models (MUSA, DUANE); a complexity analyzer and code analyzer (AdaMat); a variety of PC-based management tools, and other non- metrics tools. In summary, the Navy's program is focused on applied, tailorable and practical applications oriented metrics. Although the implementation phase is monitored and evaluated, initial life cycle phases are strongly emphasized. The collection and analysis of problem trouble reports, together with statistical evaluation of them (rate of occurrence, rate of resolution, resolution to occurrence correlation) plays a critical role in predicting software quality for NUSC. The effort has potentially significant relevance to the SDI initiative and parts and processes of their program may be exported/reused directly to become cornerstones of SDI metrics/quality programs.

The submarine systems environment has several similar characteristics to SDI component systems, including austere survivability and reliability requirements. In certain cases, the environment and operating conditions may be even more severe since maintenance corrections and fault repairs during operational conditions at sea are not allowed. In many instances, reliability and MTBF's are pushed to the limit. Large megabit transmission links are non- existant to effect downloading of new or enhanced software versions as may be found in SDI space-based subsystems. The information exchange with NUSC is in its initial phase, and further technical interchanges are envisioned.

3.2 NAVAL SURFACE WARFARE CENTER - DAHLGREN

Dr. William Farr at NSWL Strategic Systems Dept. has been active on the IEEE Software Reliability Measurement Working Group Steering Committee and a contributor to the IEEE draft standard P982.1 committee. Dr. Farr has developed a package of 8 metrics models dubbed "SMERFS" (Statistical Modelling and Estimating of Reliability Functions for Software) and last reported over 50 different users. One of which, NASA-Johnson Space Flight Center, will be discussed next. (A module package with usage documents have been received--refer to Section 5.5.2).

3.3 IBM AT NASA-JOHNSON SPACE FLIGHT CENTER

David Hamilton of IBM at NASA in Houston has been a SMERFS user and reports highly satisfactory results using the SMERFS-based Schneidewind Model. He has had one years' use with the SMERFS package. It should be noted that SMERFS is a post-facto metrics application.

The IBM Space Shuttle Programs' "On Board Primary Software Reliability Prediction" effort source information was examined for insight into metrics and models used. Paraphrasing their approach - "The emphasis of this program centers on providing good software via the removal of failure mode causes based on the early identification of design flaws at the 'front-end' of the process (i.e., requirements & design) rather than 'defense (e.g., redundancy)' against them." A key IBM approach to detecting software errors is to use Statistical Modelling of detection history data in a configuration management database. The methodology is to apply the SMERFS, a multi-model interactive computer program, to appropriate discrepancy data in the database. Model validation centers on predicting reliability. Much of the data used for analysis is derived from previously developed software that is repeatedly analyzed, catalogued and statistically analyzed. The SMERFS models are then used to find corresponding "form and fit representations". The model that best fit their data is the Schneidewind Model which uses exponentially decreasing error detection rates and a poisson process failure detection.

The validity of this approach, apart from a front-end emphasis, depends upon:

a. the application of software engineering methodologies;

b. well-defined processes;

c. performing labor intensive and extensive multi-path activities (e.g., code to function/vice-versa correlation, backward chaining analysis)

d. "(a) software error history sufficiently analyzed to allow application of (the) reliability model".

Note: In the domain of SDI software, c. and d. are not practical due to the amount and complexity of the software, and the lack of historical data available relative to it. Many of the SDI software applications are being developed for the first time.

4. CAPABILITIES, LIMITATION AND DEFICIENCIES OF MODELS

The prioritization of SDS software metrics requirements appeared to be high for Reliability, Survivability, Integrity, and Efficiency software attributes. There was moderate or medium priority for Maintainability, Portability, Usability, and Reusability, in that order. The assessment of the software metrics maturity presents a different ordering: Efficiency, Reliability, Maintainability begin most matured, with considerable less for Survivability, and negligible, if any, support for Portability, Usability and Reusability attributes. The rest of this section will summarize the capabilities of the field of software metrics models to support each of these areas.

4.1 GENERAL LIMITATIONS

[CONT86] makes some general caveats after introducing the notion of a (composite) "Quality of Software" metric (1.2.2). The book gives the following warnings:

a. Like items must be measured and compared together "apples with apples".

b. When metrics are moved from one environment to another, they should be re-calibrated.

c. Metrics are to support not replace management savvy.

d. Software metrics are poor personnel evaluators.

e. Any metric model output is not better than its input.

f. The metrics program cost should be less than its benefits.

4.2 RELIABILITY MODELS

Both the Time Between Failures (TBF) and the Failure Count models have had considerable evaluation and study. Recently, John Musa stated that failure count and time was the correct basis for reliability assessment and prediction as opposed to fault or defect counting and projection *[MUSA8903]*. However, as stated in *[IDA P-2132, 7.3]* "Current software reliability technology suffers from some fundamental problems and limitations". This paper states that the convenient adoption of hardware reliability measures obscures the fundamental difference between hardware and software reliability: that though hardware is subject to physical aging, software is not. Fresh analysis into the nature of software reliability and trustworthiness by Parnas suggest that a combinatorial/probalistic indicator may be needed: the probability that no critical fault remains.

[IDA P-2132] also raises doubts that studies based upon non-critical fault rates in software targets of average reliability are deterministic toward predictions of criticality in highly reliable software. The IDA paper goes on to quote Goel at the IDA Testing and Evaluation Workshop that available approaches are too simplistic, and they cast the very pessimistic statement "Because of the fundamental limitations of current software reliability assessment methodologies ... further work on enhancing current methodolgies (sic) is unlikely to yield satisfactory results." (Assuming that the Parnas criteria is used -- determination of the probability of a failure-causing fault.)

This pessimistic cast of the IDA paper was presaged back in 1986 by Abdel-Ghaly, Chan and Littlewood, in their review of ten reliability models. The authors pointed out that predictive quality is the only consideration for a user of software reliability models and a good "model" is not sufficient to produce good predictions. They found that the inference rules and the predictive procedure were significant to the relative success of the various models *[ABDE8609]*.

The authors present an approach for validating candidate models for a specific application. The results showed that there was no single "best buy" among them. The ten models were: Jelinski - Moranda, Bayesian Jelinski-Moranda, Littlewood, Bayesian Littlewood, Littlewood-Verral, Keiller-Littlewood, Weibull Order Statistics, Duane, Goel-Okumoto Model, and Littlewood NHPP.

4.3 COMPLEXITY/MAINTAINABILITY

More recent analysis of maintainability predictors using measures as program specification change rates, programmer's skill levels, and volume of design documentation, has considerable effect on error rates. Both *[TAKA8901]* and *[GIBS8903]* showed that structural differences do impact programmer's performance. Specifically, system improvements result in considerable improvements in programmer's performance. Gibson and Senn's study investigated this by asking three experienced professional programmers to perform three maintenance tasks on three functionally equivalent, but different in complexity, versions of a COBOL system.

Six metrics were used in this study: the Halstead's E, McCabe's cyclomatic complexity, Woodward's K, Gaffney's Jumps, Chen's MIN (Maximum Intersects Number), and Benyon-Tinker's Cx. The metrics reflected both the improvements in the system (in terms of ease in maintenance as system complexity is decreased) as well as in programmer maintenance performance. The Halstead's E, McCabe's Cyclomatic Complexity, Woodward's K, and Gaffney's Jumps reflect a decrease in complexity with improvement in system structure (control flow complexity). The Chen's MIN and Benyon-Tinker's Cx on the other hand reflect the offsetting increase in complexity due to IF nesting and number of modules.

This study was consistent with earlier work by Evangelist and work by Basili, Selby and Philips, investigating the validity of complexity measures as independent predictors of effort and or psychological complexity *[EVAN84, EVAN83, BASI83]*. These investigators concluded that the metrics by themselves were no better than lines-of-code, but when the personnel being studied was held constant (single programmer studied across multiple projects), there appeared to be some relative correlation with actual work and times to complete.

Since the metrics have shown to be meaningful in studies where the work object is changed, but the programme~ performing the work is fixed, further work studying programmer variables is needed. A detailed problem solving and work trait profiling system may be possible to account for significant programmer differences, and be a basis to increase the validity of complexity metrics for maintainability predictions.

4.4 EFFICIENCY MODELS

The use of modeling to predict sequential software efficiency is quite mature. Typically a simulation language is used to model the intended software achitecture. As development proceeds, more data is fed into the simulator, and the accuracy of the predictions improves. General purpose simulators like Simscript and GPSS have been available for many years, as have texts and articles *[ADKI7704], [SCHN7704].*

However, as pointed out by the IDA team in their review of testing technology available for concurrent and real-time programs, the methodology is relatively new. One recent technology is the *TAGS* Simulation Compiler offered by Teledyne Brown. Another is *Auto-G* from Advanced Systems Architectures of England. A third is *Statemate* by "iLOGIX", an Israeli vendor.

However promising these packages are, they require further investigation and validation.

4.5 EFFICIENCY/PRODUCTIVITY MODELS

Halstead's original "Software Science E" was a model to forecast programming effort based upon complexity and size work parameters. The *COCOMO* and *SECOMO* models are based on lines of code and other intermediate cost drivers. These models have been in use on many programs and should be considered mature *[BOEH81]*, however, David Card and others have found that the factors of change in the software maintenance environment may not be so predictable.

The maintenance cost model was studied in *[CARD8807]*. The results showed that the standard maintenance cost models based on lines of code measures proved inadequate in estimating maintenance cost. This is due to the different productivity rates of maintenance and the accumulation effect of program changes.

4.6 INTEGRITY/SECURITY MODELS

The existence of an integrity model is not obvious, but both the Air Force and Navy have developed systems security risk assessment methods *[NEUG85, pg. 4-74, 75]*, which may be used to assess system integrity factors in terms of annual levels of loss. Both methods are fairly inaccurate, using "fuzzy-metrics" (approximated orders of magnitude). This concept of fuzzy metrics may have some application to the assessment of integrity as may other elements of the security models: vulnerability inventories, threat catalogues, service and data asset valuation, and time-based exposure projections. Fairly recent research into fuzzy systems *[NEGO81]* and modeling *[NEGO87]* theory has developed the mathematical foundations needed to refine such applied risk models, but much more study is needed.

4.7 PORTABILITY, USABILITY, REUSABILITY

Models to support assessments of portability, usability and reusability are not available. This realization was also derived by the SPARTA team [SPART8903]. These attributes were ranked as medium in the aggregate rankings, so some assessment of their relative value for research and development is warranted.

4.8 COMPOSITE METRICS

Though both *[CONT86]* and *[RADC878A]* present a grand accumulation formula for assessing the many quality factors and arriving at a signle coefficient, there is considerable skepticism that such a calculation would obscure the meaningfulness to seasoned managers. Another method is outlined in *[GOIC81]* and *[SING87]* -- Multicriteria Modeling and/or Fuzzy Multicriteria Modeling.

[SING87, pg. 1827] presents a good introductory example. For a software quality metrics application refer to the discussions about the Naval Underwater Systems Command.

5. AVAILABLE SOFTWARE METRICS SUPPORT

This section presents the Subtask 3 survey, features assessment and recommendation of available software metrics tools and environments.

5.1 EXISTING TOOLS AND ENVIRONMENTS

An extensive list of tools and environments have been extracted from TBE's TASQ database for examination. A more extensive list of tools and environments reviewed and others that are under review is contained in Appendix C.

From this task's point of view, two categories of tools have been established: tools directly supporting a metric (e.g., McCabe, Halstead, Complexity) and ancillary support tools (i.e., tools that can support or assist in the collection or analysis of a metric). Ancillary support tools consist of the following categories:

- Program Management & Support Tools
- Computer-Aided Software Engineering Tools
- Linkers, Loaders, Debuggers
- Test Generators
- Spreadsheets
- Databases
- Environments

Thus, while spreadsheets such as LOTUS 1-2-3 can be extremely useful in collecting and classifying metrics and measurements, spreadsheets are not considered metrics tools.

5.2 ANALYTIC EVALUATION OF TOOLS/ENVIRONMENTS

A number of metrics tools have been identified:

Tool	Source
SMART	USA-CECOM/TBE
SMERFS	NSWC-Dahlgren/Wm. Farr
McCabe Metric	Intermetrics
Halstead Metric	Intermetrics
Complexity Measures Tool	EVB
Complexity Metric	Computer Systems Design
Analyze	Autometric, Inc.
Complexity Measure	PD SIMTEL 20
ADADL	Software Systems Design, Inc.
Byron	Intermetrics
Analyze	AdaSoft
Adamat	Dynamics Research Corp.
Complexity Analysis Tool	McCabe Associates
Ada Complexity Analysis Tool (ACAT), (McCabe)	Teledyne Brown Engineering
AdaPIC	Arcadia Consortium
Logiscope	Verilog
COCOMO	TRW-Redondo Beach
SECOMO	RADC-DACS

With the emphasis for the Strategic Defense Initiatives to capitalize on modern software engineering methods and approaches, and the Ada Technologies initiatives, only a few of these tools will be discussed further.

5.2.1 Software Management and Reporting Tool (SMART)

The SMART package is targeted for deployment at the U.S. Army Communications - Electronics Command (CECOM) July-August 1989. The first program it will be applied on is the Advanced Field Artillery Tactical Data System (AFATDS).

The software metrics goal is to implement the Software Management and Quality Indicators as per AFSCP-800-43 and AFSCP-800-14 in an efficient, extensible architecture. The software metrics indicators and data management will be based on IBM PC-compatible machines using the popular "dBase3" formats with the "Clipper" data base language system.

Table 5-1 shows some of the data tracked by SMART.

TABLE 5-1
OVERVIEW OF SMART SOFTWARE DATA

TYPE INDICATOR
Deviations
Waivers
Software Trouble Reports
Test Incident Forms
Engineering Change Proposals
Software Improvement Reports
Software Discrepancy Reports
Computer Resource Utilization
Software Development Manpower
Requirements Definition and Stability
Software Progress - Development and Test
Cost/Schedule Deviations
Software Development Tools
Completeness
Design Structure
Defect Density
Fault Density
Test Coverage
Software Maturity
Documentation

TABLE 5-2
LIST OF SAMPLE GRAPHICS FOR SMART

COMPUTER RESOURCE UTILIZATIONS (3)
- Primary Memory
- Secondary Memory
- CPU Utilization
- I/O Utilization

SOFTWARE DEVELOPMENT MANPOWER (WBS)

REQUIREMENTS DEFINITION AND STABILITY (2 LEVELS)
- System; CSCI

DEVELOPMENT AND TEST
- Scheduled/Actual; CSCI & CSC

COST/SCHEDULE DEVIATIONS

SOFTWARE DEVELOPMENT TOOLS

COMPLETENESS
- Requirements %/month
- Implementation %/month

DESIGN STRUCTURE
- Modularity
- Complexity/Dependency

DEFECT DENSITIES
- Requirements
- Design
- Code

FAULT (FAILURE) DENSITY

TEST COVERAGE

SOFTWARE MATURITY

DOCUMENT TROUBLES (BY MODULE)

5.2.2 Statistical Modeling and Estimation of Reliability Functions for Software (SMERFS)

SMERFS was developed several years ago as an aid in the evaluation of software reliability. In its original design it was targeted for mainframe and mini-computer environments. Since then it has also been adapted to operate on micro-computers, specifically IBM-PC/XT compatibles.

The current version of SMERFS has incorporated eight software reliability models. The models include the following: (1) Musa's Execution Model, (2) Goel-Okumoto Non-Homogeneous Poisson Model, (3) Adapted Goel-Okumoto Non-Homogeneous Poisson Model, (4) Moranda's Geometric Model, (5) Schafer's Generalized Poisson Model, (6) Schneidewind's Model, (7) Littlewood-Verrall Bayesian Model, and (8) the Brooks-Motley Model.

SMERFS contains a driver which is claimed to make it machine independent. The driver is a subset of the American Standards Institute (ANSI) specifications for the FORTRAN 77 compiler. Several user selectable options are available within the driver and allow the system to be configured to produce: better predictions; output plots and catalogued output files. Currently SMERFS is operational on three main computer groups at the Naval Surface Weapons Center (NSWC), Dahlgren, VA. The three computer groups include the CDC CYBER 170/875, the Vaxcluster 11/785, and a large number of IBM-compatible PCs. Dr. William H. Farr, of NSWC, and Mr. Oliver D. Smith, of EG&G Washington Analytical Services Center, Inc. both claim that transferring SMERFS to other computers should be very easily accomplished. [FARR8812]

Besides containing operating instructions within its interactive mode, two additional pieces of documentation are available for use with SMERFS. The two supplemental reports are: (1) SMERFS Library Access Guide (*NSWC-TR-84-371, Rev. 1*), and (2) SMERFS User's Guide (*NSWC-TR-84-373, Rev. 1*). These two publications allow a potential user to preview the system. Examples are provided throughout the User's Guide, allowing a potential user to acquire an overview of the SMERFS processing. In addition, the guide also shows actual software reliability analyses performed on the CDC CYBER 170/875.

The SMERFS systems show tremendous potential for use in the SDI environment with a minimum of modification.

5.2.3 McCabe Complexity Tools

Tools based on McCabe Complexity Analysis are of limited use in the Ada domain,unless SDI applications use older implementation languages (e.g., COBOL, CMS-2, FORTRAN). Technical Report *MC87- McCabe II-0003*, Extending McCabe's Cyclomatic Complexity Metric for Analysis of Ada Software, *[TBE8703]* U.S. Army AMCCOM; Dover, N.J. under contract DAAA21-85-D-0010 identifies extensions that are required of McCabe's Theorems if Ada, or other modern language is the implementation language. McCabe complexity applications are limited to pre-Ada higher order languages that do not contain modern software engineering abstraction process or language constructs such as packages, generics or tasking mechanisms. Thus, pre-Ada tools that fall into this category (which is most of them) will require updating.

The above referenced report has resulted in technical report MC87-McCabe II-0005, Modified A-Level Software Design Specification for the Ada Complexity Analysis Tool which Automates the Extended McCabe's Cyclomatic Complexity Metric, *[TBE8704]* U.S. Army

AMCCOM; Dover, N.J., contract DAAA21-85-D- 0010, establishing the requirements for an Ada Complexity Analysis Tool (ACAT) to support Ada implementations.

A number of Ada based or derived program design language (*PDL*) analysis tools have emerged over the past several years. Two of those listed, *Byron* and *ADADL*, are mature enough to be employed to perform code analysis completeness and consistency checking, cross referencing, structure checking, code verification completeness and correctness, and interface analysis.

ADADL has both a McCabe and an ADADL complexity metric for both pseudocode and the Ada code for each program unit. Both of these complexity metrics have not been evaluated for degree of appropriateness.

These are estimation metrics tool used on pseudocode or Ada (late life cycle phase application). It should also be noted that a standard DoD Ada PDL does not and may never exist, thus PDL tool invocation is at the implementers or applications level discretion. Automatic code generation or a shift in design emphasis to an *SADMT [IDA8804A&B]* or like representation may obviate the need for such in the future.

5.2.4 Software Engineering Cost Model (SECOMO)

SECOMO is an interactive software cost estimation tool, based on the *COCOMO* cost model, for calculating the total technical and support manpower requirements of a Life Cycle Software Engineering (LCSE) Center. SECOMO is maintained and distributed by the Rome Air Development Centers Data and Analysis Center for Software (DACS) at a $50 charge.

SECOMO includes a "Care" cost limit for the fire-up phase of an LCSE Center.

Developed and maintained by the IIT Research Institute and RADC, it is kept current with the TRW/Barry Boehm COCOMO users days recommendations.

Significant enhancements are:

- An Ada parameters set
- Pull down menus and user efficiencies
- Site-timing parameters

Army Materiel Command sites which are user include:

- CECOM
- AVSCOM
- AMCCOM
- MICOM

Navy sites include: NSWC-Dahlgren, NUSC - New London, NCSC - Panama City, FL; NTS - Orlando.

5.2.5 Revised Enhanced Version in COCOMO "REVIC"

REVIC is another COCOMO enhancement maintained by an Air Force Center. The headquarters command at Kirtland AFB, New Mexico has developed an extensive set of project performance data which they use to tune the REVIC model. Updates and enhancements to baseline COCOMO and to SECOMO are brought into REVIC. (No Charge) HQCMP/EPR, Kirtland AFB, NM 87117-5000.

The REVIC User Group "RUG" is chaired by Dr. George Hozier at the University of New Mexico.

The active users include many Air Force sites and contractors:

Air Force Commands
ESD, RADC, ACS-Pentagon

Contractors
The Aerospace Corporation
Boeing
Hughes
GE
TRW
Lockheed
Textron

5.3 EXPERIENCE ASSESSMENTS

Adamat, both versions of Analyze, and Logiscope are mature metrics tools for extensive complexity analysis. Logiscope, at this time, does not support Ada source code, the others do. Adamat appears to have (based on literature review) the most extensive set of metrics criterion. A partial list is identified:

Anomaly Management: Default initialization, basic loops containing a conditional exit or return, strong type checking and constraint checking, raising user defined exceptions, range checking, etc.

Independence: Isolation of input/output routines, isolation of tasking statements and declarations, independence from system- dependent modules, access types, package system, use of length clauses, etc.

Modularity: Use of private and limited private types, single type declarations in package specifications, etc.

Self-Descriptions: Use of comments, use of identifier names, etc.

Simplicity: Boolean expressions, labels, decision points, branches, branch constructs, nesting levels, use of literals, procedure calls, etc.

System Clarity: Parenthesized expressions, no default mode parameters, access types declared private, loops-modules-blocks names, etc.

The TASQ and ISEC [TBE8609] tools/environment databases are extensive and robust (containing several thousand tools). Initial metrics queries have resulted in a sparse tool identification. A formal and more extensive categorization will be completed by mid-April, and will be categorized per the matrix contained in Appendix D for completeness. All tools identified in Appendix C will also be matrix categorized as such. A metrics tools expansion list much beyond that identified in this section is not expected. Recent dialogue with Ada environment developers and tool vendors at such exhibits/expositions as TRIADA, 7th Annual National Ada Conference, CASE EXPO, NSIA, etc., reveal that metrics tools are sparse and not the focus of their development efforts. This is understandable in light of the state and maturity of Ada technology. The latter may account for the individual DoD service initiatives aimed at software quality programs and tool initiatives such as those identified in this report and focusing on metrics. The SDI will in all probability require its own tailored metrics and quality control program similar to those of NUSC, NASA and AMMCOM. Much of the present developer/vendor effort is directed at environments, compiler maturation and efficiency, and related support tools (e.g., linkers, loaders, editors).

Many of the Ada developers at this time are looking for third party vendors to provide them with complementary/synergistic metrics tools (Rational and Alsys are examples of such).

The emphasis on predictive metrics and early life cycle phase emphasis has identified major metrics tool deficiencies in these areas - virtually non-existent. Well defined metrics, measurand relationships and formalisms in the early phases supported by tools have yet to appear to provide the productivity impact they are expected to have. The AdaPIC tool set (a futures project) ongoing within the Arcadia consortium, holds some promise, but these new tools have yet to be developed [*WOLF8903*]. Similarly, TBE's ACAT is under development.

Unfortunately many of the AdaPIC tools will not address the early life cycle phases, but are aimed at complementing the emerging development environment. Specifically, more systems engineering metrics tools are required to support system synthesis and couple with the early software engineering processes.

5.4 REQUISITE ENVIRONMENT/TOOLSET FEATURES

As stated in Section 5.3, the proposed TASQ format (identified in the matrix of Appendix D) will be utilized to provide the next elaboration update of tools contained in Appendix C.

REFERENCES

REF.1
ALPHABETICAL DOCUMENT INDEX

[ABDE8609] Abdel-Ghaly, Chan & Littlewood; Evaluation of Competing Software Reliability Predictions; IEEE Transactions, 9/86.

[ADKI7704] Adkins & Pooch; Computer Simulation: A Tutorial; Computer, 4/77.

[ALBR8411] Albrecht, A.J.; AD/M Productivity Measurement and Estimate Validation; IBM CIS&A Guideline 313; 11/1/84.

[ALBR8311] Albrecht & Gaffney; Software Function, Source Lines of Code and Development Effort Prediction: A Software Science Validation; IEEE Transactions, 11/83.

[ARMY8408] U.S. Army/Computer Systems Command; Software Quality Engineering Handbook; Subcontract No. 7202, 8/84.

[ARNO86] Arnold, Robert S.; Tutorial: Software Restructuring; IEEE Computer Society Press, 1986. Includes [YAU8011], [HARR8209]

[BALZ8311] Balzer, Cheatham & Green; Software Technology in the 1990's: Using a New Paradigm; Computer, 11/83.

[BASI8311A] Basili & Hutchens; An Empirical Study of a Syntactic Complexity Family; IEEE Transactions, 11/83.

[BASI8709] Basili & Rombach; Implementing Quantitative SQA: A Practical Model; IEEE Transactions, 9/87.

[BASI8703] Basili & Rombach; TAME: Tailoring an Ada Measurement Environment; Proceedings from 5th National Joint Ada Conference, 3/87.

[BASI8311B] Basili, Selby, Phillips; Metric Analysis & Data Validation Across Fortran Projects; IEEE Transactions; 11/83.

[BEHR8311] Behrens, Charles; Measuring the Productivity of Computer Systems Development Activities with Function Points; IEEE Transactions, 11/83.

[BEIZ84] Beizer, Boris; Software System Testing and Quality Assurance; Van Nostrand Reinhold Co., 1984.

[BELZ8603] Belz, Frank C.; Applying the Spiral Model: Observations on Developing System Software in Ada; Proceedings from 4th National Ada Conference, 3/86.

[BEND8703] Bendifallah & Scacchi; Understanding Software Maintenance Work; IEEE Transactions, 3/87.

[BOEH81] Boehm, Barry W.; Software Engineering Economics; Prentice-Hall, New Jersey, 1981.

[BOEH8810] Boehm & Papaccio; Understanding and Controlling Software Costs; IEEE Transactions, 10/88.

[BOWE8610] Bowen, John B.; Application of a Multi-Model Approach to Estimating Residual Software Faults and Time Between Failures; Quality & Reliability Engineering International, Vol. 3; 10/22/86.

[BRIT8811] Britcher, R.N.; Using Inspections to Investigate Program Correctness; Computer, 11/88.

[BROW8407] Browne, J.C.; Understanding Execution Behavior of Software Systems; Computer, 7/84.

[CARD8508] Card, Page & McGarry; Criteria for Software Modularization; Proceedings IEEE Conference on Software Engineering; 8/85. (see SEI section)

[CARD8602] Card, Church & Agresti; An Empirical Study of Software Design Practices; IEEE Transactions; 2/86. (see SEI section)

[CARD8707] Card, McGarry & Page; Evaluating Software Engineering Technologies; IEEE Transactions, 7/87.

[CARD8709] Card, Cotnoir, Goorevich; Managing Software Maintenance Cost & Quality; Proceedings IEEE Conference on Software Maintenance, 9/87. (see SEI section)

[CARD8812] Card & Agresti; Measuring Software Design Complexity; Journal of Systems and Software, 12/88. (see SEI section)

[CARD87] Card & Agresti; Resolving the Software Science Anomaly; Journal of Systems and Software, 1987. (see SEI section)

[CARD8807] Card, David; The Role of Measurement in Software Engineering; Proceedings 2nd IEE/BCS Software Engineering Conf., 7/88. (see SEI section)

[CARR8603] Carrio, Miguel A.; The Technology Life Cycle and Ada; Proceedings from 4th National Ada Conference, 3/86.

[CHAL87] Chalam, V.V.; Adaptive Control Systems - Techniques & Applications; Marcel Dekker, Inc., 1987.

[CHEU8706] Cheung & Li; An Empirical Study of Software Metrics; IEEE Transactions, 6/87.

[CHIE8607] Chien, Y-T; Expert Systems in the SDI Environment; Computer, 7/86.

[CHUB89R] Chubb, Bruce; Lessons Learned in the Setup and Management of Productivity Improvement Methods.

[CINL75] Cinlar, Erhan; Introduction to Stochastic Processes; Prentice-Hall, Inc. 1975.

[COHE85] Cohen, Paul R.; Heuristic Reasoning about Uncertainty: An Artificial Intelligence Approach; Pitman Publishing, Inc., 1985.

[CONT86] Conte, Dunsmore, Shen; Software Engineering Metrics & Models; Benjamin/Cummings Publishing Co.; 1986.

[DAVI8810] Davis, Bersoff, Comer; A Strategy for Comparing Alternative Software Development Life Cycle Models; IEEE Transactions, 10/88.

[DAVI8809] Davis & LeBlanc; A Study of the Applicability of Complexity Measures; IEEE Transactions, 9/88.

[DELO8807] DeLoach, Scott A.; An Interface-Based Ada Programming Support Environment; ACM Press-Ada Letters; July/Aug. 1988.

[DEMU8807] Demurjian & Hsiao; Towards a Better Understanding of Data Models Through the Multilingual Database System; IEEE Transactions, 7/88.

[DOD8807A] Department of Defense; Report of the Defense Science Board Task Force on Military Software-9/87; ACM Press-Ada Letters; July/Aug. 1988.

[DOD8807B] Department of Defense; Ada Board Response to Report of Defense Science Board Task Force on Military Software-2/88; ACM Press-Ada Letters; July/Aug. 1988.

[DORN75] Dorny, C. Nelson; A Vector Space Approach to Models & Optimization; John Wiley & Sons; 1975.

[DUNS8805] Dunsmore, H.E.; Evidence Supports some Truisms, Belies Others; Quality Time-IEEE Software, 5/88.

[ELLI88] Ellis & Wygant; A System-Oriented Methodology to Support Software Testability; ITEA Journal, 1988.

[EVAN84] Evangelist, Michael; An Analysis of Control Flow Complexity; COMPSAC Proceedings, 1984.

[EVAN83] Evangelist, Michael; Software Complexity Metric Sensitivity to Program Structuring Rules; Journal of Systems & Software; 1983.

[FARR88] Farr, Smith & Schimmelpfenneg; A PC Tool for Software Reliability Measurement; Proceedings from Institute of Environment Sciences; 1988.

[FARR8812A] Farr & Smith; Statistical Modeling and Estimation of Reliability Functions for Software (SMERFS) User's Guide; NSWC-TR-84-373, 12/88.

[FARR8812B] Farr & Smith; Statistical Modeling and Estimation of Relability Functions for Software (SMERFS) Library Access Guide; NSWC-TR-84-371, 12/88.

[FARR8309] Farr, William H.; A Survey of Software Reliability Modeling and Estimation; NSWC-TR-82-171; 9/83.

[FREE87] Freeman, Peter; Tutorial: Software Reusability; IEEE Computer Society Press, 1987.

[GADO88] Gados, Ronald; Guidelines for Certification of Strategic Defense System Simulation Models; ITEA Journal, 1988.

[GELP8806] Gelperin & Hetzel; The Growth of Software Testing; Communications of the ACM, 6/88.

[GIBS8903] Gibson & Senn; System Structure and Software Maintenance Performance; Communications of the ACM, 3/89.

[GOIC89RA] Goicoechea, Burden, Sanchez; Evaluation & Ranking of Alternate Systems with Ariadne: An Application to the Economy Car Selection Problem. (see GMU Info. section)

[GOIC81] Goicoechea, Hansen & Henrickson; A Hierarchical-Multilevel Approach to the Development & Ranking of Systems in Tanning Industry; Computing & Industrial Engineering, 1981. (see GMU Info. section)

[GOIC89RB] Goicoechea, Ambrose; Metrics and Trade-Off Analysis in Software Design, Production, and Evaluation.

[GOIC89RC] Goicoechea, Ambrose; Software Metrics: Their Measurement and Use in Models. (see GMU Info. section)

[HARR89RA] Harrison, Warren; Applying McCabe's Complexity Measure to Multiple-Exit Programs. (see PSU Info. section)

[HARR8209] Harrison, Magel, Kluczny, DeKock; Applying Software Complexity Metrics to Program Maintenance; Computer Magazine, 9/82. (see PSU Info. section)

[HARR86] Harrison & Cook; Are Deeply Nested Conditionals Less Readable?; Journal of Systems and Software, 1986. (see PSU Info. section)

[HARR89RB] Harrison & Magel; A Complexity Measure Based on Nesting Level. (see PSU Info. section)

[HARR88] Harrison, Warren; MAE: A Syntactic Metric Analysis Environment; Journal of Systems and Software; 1988. (see PSU Info. section)

[HARR87] Harrison & Cook; A Micro/Macro Measure of Software Complexity; Journal of Systems and Software; 1987. (see PSU Info. section)

[HARR8602] Harrison & Cook; A Note on the Berry-Meekings Style Metric; *Communications of the ACM, 2/86. (see PSU Info. section)*

[HARR89RC] Harrison, Warren; Software Science and Weyuker's Fifth Property. (see PSU Info. section)

[HARR8804] Harrison, Warren; Using Software Metrics to Allocate Testing Resources; Journal of Management Information Systems, Spring '88. (see PSU Info. section)

[HENR8109] Henry & Kafura; Software Structure Metrics Based on Information Flow; IEEE Transactions, 9/81.

[HUMP8703] Humphrey, Watts S.; Software Process Management; 9th International Conference on Software Engineering, 3/87.

[IDA8804A] IDA-Deliverable to SDIO; *Strategic Defense Initiative Architecture Dataflow* Modeling Technique, Version 1.5; Report # P-2035, 4/88.

[IDA8804B] IDA-Deliverable to SDIO; A Simple Example of an SADMT Architecture Specification, Version 1.5; Report # P-2036, 4/88.

[IDA8812] IDA-Deliverable to SDIO; SDS Software Testing & Evaluation: A Review of the State-of-the-Art in Software Testing and Evaluation; Report # P-2132, 12/88.

[JEFF8707] Jeffery, D. Ross; Time-Sensitive Cost Models in the Commercial MIS Environment; IEEE Transactions, 7/87.

[KAFU8703] Kafura & Reddy; The Use of Software Complexity Metrics in Software Maintenance; IEEE Transactions, 3/87.

[KARI8812] Karimi & Kuo; User Interface Design from a Real Time Perspective; Communications of the ACM, 12/88.

[KAVI8710] Kavi, Buckles and Bhat; Isomorphisms Between Petri Nets and Dataflow Graphs; IEEE Transactions, 10/87.

[LEAC8703] Leach, Ronald J.; Ada Software Metrics and Their Limitations; Proceedings from 5th National Joint Ada Conference, 3/87.

[LEE8710] Lee, Tony; An Information-Theoretic Analysis of Relational Databases--Part I: Data Dependencies and Information Metric; IEEE Transactions, 10/87.

[LEW8811] Lew, Dillon & Forward; Software Complexity and its Impact on Software Reliability; IEEE Transactions, 11/88.

[LICH8601] Lichtman, Zavdi; Generation and Consistency Checking of Design and Program Structures; IEEE Transactions, 1/86.

[MART8407] Martin & Brice; Effect of Hardware-Software Interaction on Performance; Computer, 7/84.

[MCCA7612] McCabe, Thomas; A Complexity Measure; IEEE Transactions, 12/76.

[MCCL7805] McClure, Carma; A Model for Program Complexity Analysis; Proceedings from 3rd International Conference on S/W Engineering; 5/78.

[MCKA89R] McKay, Charles; A Proposed Framework for the Tools & Rules to Support the Life Cycle of the Space Station.

[MEND8205] Mendis, Kenneth S.; Quantifying Software Quality; Computers, 5/82.

[MILL7612] Mills, Harlan D.; Software Development; IEEE Transactions, 12/76.

[MUSA8903] Musa, John D.; Faults, Failures, and a Metrics Revolution; IEEE Software, 3/89.

[MUSA87] Musa, Iannino, Okumoto; Software Reliability: Measurement, Prediction, Application; McGraw-Hill, New York; 1987.

[MUSA8902] Musa, John D.; Tools for Measuring Software Reliability; IEEE Spectrum, 2/89.

[MYER8611] Myers, W.; Can Software for the Strategic Defense Initiative Ever be Error-Free?; Computer, 11/86.

[NELS87] Nelson & Carroll; Tutorial: Fault-Tolerant Computing; IEEE Computer Society Press; 1987.

[NEUG8510] Neugent, Gilligan, Hoffman & Ruthberg; Technology Assessment: Methods for Measuring the Level of Computer Security; NSB Special Publication 500-133, 10/85.

[NEUM8609] Neumann, Peter Gabriel; On Hierarchical Design of Computer Systems for Critical Applications; IEEE Transactions, 9/86.

[NYBE89] Nyberg, Karl; Ada: Sources & Resources 1989 Edition.

[OSTR8806] Ostrand & Balcer; The Category-Partition Method for Specifying & Generating Functional Tests; Communications of the ACM, 6/88.

[PERK8703] Perkins & Gorzela; Experience Using an Automated Metrics Framework to Improve the Quality of Ada Software; Proceedings from 5th National Joint Ada Conference, 3/87.

[RAMA8808] Ramamurthy & Melton; A Synthesis of Software Science Measures and the Cyclomatic Number; IEEE Transactions, 8/88.

[RICH8808] Rich & Waters; Automatic Programming Myths and Prospects; Computer, 8/88.

[ROLA8606] Roland, Jon; Software Metrics; Computer Language, 6/86.

[ROMB8703] Rombach, H. Dieter; A Controlled Experiment on the Impact of Software Structure on Maintainability; IEEE Transactions, 3/87.

[RADC8205] Rome Air Development Center; Software Testing Measures; RADC-TR-82-135, 5/82.

[RADC8308] Rome Air Development Center; A Guidebook for Software Reliability Assessment; RADC-TR-83-176, 8/83.

[RADC8403] Rome Air Development Center; Software Test Handbook - Software Test Guidebook; RADC-TR-84-53, Vol. II; 3/84.

[RADC8502] Rome Air Development Center; Specification of Software Quality Attributes; RADC-TR-85-37, Vol. I, II, III; 2/85.

[RADC878A] Rome Air Development Center; Methodology for Software Reliability Prediction and Assessment; RADC-TR-87-171, Vol. I; 8/87.

[RADC878B] Rome Air Development Center; Software Reliability Prediction and Estimation Guidebook; RADC-TR-87-171, Vol. II; 8/87.

[SCHN8904] Schneidewind, Norman; Distributed System Software Design Paradigm with Application to Computer Networks; IEEE Transactions, 4/89.

[SCHN8703] Schneidewind, Norman; The State of Software Maintenance; IEEE Transactions, 3/87.

[SCHN7704] Schneidewind, Norman; The Use of Simulation in the Evaluation of Software; Computer, 4/77.

[SCHU8805] Schultz, Herman P.; Software Management Metrics; ESD-TR-88-001, 5/88.

[SELB8703] Selby, Richard W.; Incorporating Metrics into a Software Environment; Proceedings from 5th National Joint Ada Conference, 3/87.

[SHAT8808] Shatz, Sol; Towards Complexity Metrics for Ada Tasking; IEEE Transactions, 8/88.

[SHUM88] Shumskas, Anthony; Why Higher Reliability Software Should Result from Reduced Test and Increased Evaluation; ITEA Journal, 1988.

[SIEG8807] Siegrist, Kyle; Reliability of Systems with Markov Transfer of Control; IEEE Transactions, 7/88.

[SING87] Singh, Madan; Systems & Control Encyclopedia - Theory, Technology, Applications; Vol. 3, 6, 8; Pergamon Press; 1987.

[SLON8703] Slonaker, Smith, Prizant & Giles; Development of Multi-Tasking Software in Ada - a Case Study; Proceedings from 5th National Joint Ada Conference, 3/87.

[SPAR8808] SPARTA, TBE & TASC Deliverable to SDIO; Task Order 5 BM/C3 Architectures - (Tools) Subtask 1 Architectural Representation Methodology; Contract No. SDIO84-88-C-0018, 8/15/88.

[SRIV8801] Srivastava & Farr; The Use of Software Reliability Models in the Analysis of Operational Software System; Proceedings from the Institute of Environmental Sciences, 1/13/88.

[STAN85] Stankovic, John A.; Reliable Distributed System Software; IEEE Computer Society Press; 1985.

[STEL8807] Stelovsky & Sugaya; A System for Specification and Rapid Prototyping of Application Commanc Languages; IEEE Transactions, 7/88.

[TAKA8901] Takahashi & Kamayachi; An Empirical Study of a Model for Program Error Prediction; IEEE Transactions, 1/89.

[TBE8609] Teledyne Brown Engineering Deliverable to ISEC; Generic Tools Requirements for ISEC Ada Transition; MJ86-85-C0244, Contract No. F49642-85-C0244, 9/30/86.

[TBE8703] Teledyne Brown Engineering Deliverable to AMCCOM; Extending McCabe's Cyclomatic Complexity Metric for Analysis of Ada Software; MC87-McCabe II-0003, Contract No. DAAA21-85-D-0010, 3/87.

[TBE8704] Teledyne Brown Engineering Deliverable to AMCCOM; Modified A-Level Software Design Specification for the Ada Complexity Analysis Tool Which Automates the Extended McCabe's Cyclomatic Metric; MC87-McCabe II-0005; Contract # DAAA21-85-D-0010, 4/87.

[TBE8710] Teledyne Brown Engineering Deliverable to CECOM; Software Methodology Catalog; MC87-COMM/ADP-0036, Contract No. DAAB07-86-D-R001, 10/87.

[TBE8808A] Teledyne Brown Engineering Deliverable to SDIO; BM/C3 Architecture Tools, Tool Set Recommendation - TAGS and SADMT and Software Test Plan for the SADMT-Based IORL Simulation Compiler; Contract No. SDIO84-88-C-0018, 8/19/88.

[TBE8808B] Teledyne Brown Engineering Deliverable to SDIO; Technical Report for the Development of a IORL Software Translator to SADMT; Contract No. SDIO-84-88-C-0018, 8/26/88.

[TBE8811] Teledyne Brown Engineering Deliverable to CECOM; Implementation Guidelines for Software Management & Quality Indicators; Contract No. DAAB07-86-D-R001, Draft-9/88, Revised-11/88.

[TENN8809] Tenny, Ted; Program Readability: Procedures Versus Comments; IEEE Transactions, 9/88.

[TITA8809] Titan Systems Deliverable to SDIO; Preliminary Software Test and Evaluation Assessment and Recommendation; 9/30/88.

[TOHM8903] Tohma, Tokunaga, Nagase & Murata; Structural Approach to the Estimation of the Number of Residual Software Faults Based on the Hyper-Geometric Distribution; IEEE Transactions, 3/89.

[TRAC88] Tracz, Will; Tutorial: Software Reuse: Emerging Technology; IEEE Computer Society Press; 1988.

[WEID8602] Weiderman, Nelson; Evaluation of Ada Environments; Report # SEI-86-MR-2, 2/5/86.

[WEIS8810] Weiss & Weyuker; An Extended Domain-Based Model of Software Reliability; IEEE Transactions, 10/88.

[WEYU8809] Weyuker, Elaine; Evaluating Software Complexity Measures; IEEE Transactions, 9/88.

[WEYU8806] Weyuker, Elaine; The Evaluation of Program-Based Software Test Data Adequacy Criteria; Communications of the ACM, 6/88.

[WOLF8903] Wolf, Clarke & Wileden; The AdaPIC Tool Set: Supporting Interface Control and Analysis Throughout the Software Development Process; IEEE Transactions, 3/89.

[WOOD79] Woodward, Hennell & Hedley; A Measure of Control Flow Complexity in Program Text; IEEE Transactions, 1979.

[WU8703] Wu, Basili & Reed; A Structure Coverage Tool for Ada Software Systems; Proceedings from 5th National Joint Ada Conference, 3/87.

[YAU8808] Yau, Nicholl, Tsai, Liu; An Integrated Life-Cycle Model for Software Maintenance; IEEE Transactions, 8/88.

[YAU8011] Yau & Collofello; Some Stability Measures for Software Maintenance; IEEE Transactions, 11/80.

[YAU8606] Yau & Tsai; A Survey of Software Design Techniques; IEEE Transactions, 6/86.

[YU8809] Yu, Shen, Dunsmore; An Analysis of Several Software Defect Models; IEEE Transactions, 9/88.

REF.2 CATAGORY INDICES

METRICS PROGRAMS

1. NUSC - Navy Submarine Combat System Applied Software Metrics Program - 3/15/89
2. IBM - Space Shuttle Programs - 7/29/88
3 USAF R&M 2000 Program - 1/1/89
4. NSWC - Statistical Modeling & Estimation of Reliability Functions for Software (SMERFS) - Dahlgren, VA, 12/88

SDS POLICIES AND PROCEDURES

1. Strategic Defense System Software Policy. Draft 7/19/88

TECHNOLOGY FOR ASSESSMENT OF SOFTWARE QUALITY (TASQ)

1. Database Reports
2. AMC S/W Task Force Brief - 12/14/88
3. POC Cross Reference Draft Report to AMCCOM; Contract No. DAAA21-88-D-0012, Technical Report MC89-TASQ-0003, 1/27/89

SOFTWARE ENGINEERING INSTITUTE (SEI) (received from)

1. The Role of Measurement in Software Engineering; David Card; Proceedings 2nd IEE/BCS Software Engineering Conf., 7/88 [CARD8807]
2. Resolving the Software Science Anomaly; Card & Agresti; Journal of Systems & Software, 1987 [CARD87]
3. Measuring Software Design Complexity; Card & Agresti; Journal of Systems & Software, 1988 [CARD88]
4. Managing Software Maintenance Cost & Quality; Card, Cotnoir, Goorevich; Proceedings IEEE Conference on Software Maintenance, 9/87 [CARD8709]
5. An Empirical Study of Software Design Practices; Card, Church, Agresti; IEEE Transactions; 2/86 [CARD8602]
6. Criteria for Software Modularization; Card, Page, McGarry; Proceedings IEEE Conf. on Software Engineering; 8/85 [CARD8508]

INSTITUTE FOR DEFENSE ANALYSES (IDA)

1. SDS Software Testing & Evaluation: A Review of the State-of-the-Art in Software Testing and Evaluation; IDA Deliverable to SDIO, Report # P-2132, 12/88 [IDA8812]
2. Strategic Defense Initiative Architecture Dataflow Modeling Technique; IDA Deliverable to SDIO, Report P-2035, 4/88 [IDA8804A]
3. A Simple Example of an SADMT Architecture Specification; IDA Deliverable to SDIO, Report P-2036, 4/88 [IDA8804B]

MIL-STDS - IEEE/ANSI/ACM/IDA/Directives/Guidelines/etc.

1. AFSCP-800-14 S/W Quality Indicators 1/20/87
2. AFSCP-800-43 S/W Management Indicators 1/31/86
3. IEEE Std. for Software Productivity Metrics 9/1/88
4. IEEE P1061 Std. for Software Quality Metrics, Draft 2/23/87 & 11/17/88
5. IEEE P982.1/D3.0 Std, A Standard Dictionary of Measures to Produce Reliable Software, Draft 5/88
6. IEEE P982.2/D6, Draft Guide for the Use of Standard Dictionary of Measures to Produce Reliable Software, 5/88
7. IEEE P1044/D3 Draft Standard of a Standard Classification for Software Errors, Faults & Failures, 12/87
8. Orlando II, Panel V, Management Indicators and Quality Metrics, 1/30/87
9. AR 1000-1 Department of the Army, Basic Policies for Systems Acquisition, 5/1/81
10. Draft DoDD 5000.29, Management of Computer Resources in Major Defense Systems, 1/15/86
11. DoDD 3405.1, Computer Programming Language Policy, 4/2/87
12. DoDD 3405.2, Use of Ada in Weapon Systems, 3/30/87
13. Draft DoDD 3405.xx, Computer Software Policy, 10/29/85
14. MIL-STD-2167A, Defense System Software Development, 2/29/88

GEORGE MASON UNIVERSITY

1. Tutorial paper: A User-Oriented Listing of Multiple Criteria Decision Methods; Despontin, Moscarola, Spronk; Belgian Journal of Statistics
2. Metrics & Trade-Off Analysis in Software Design, Production, & Evaluation; Ambrose Goicoechea, GMU [GOIC89RB]
3. A Hierarchical-Multilevel Approach to the Development & Ranking of Systems in Tanning Industry; Goicoechea, Hansen & Henrickson, Computing & Industrial Engineering, 1981 [GOIC81]
4. Evaluation & Ranking of Alternate Systems with Ariadne: An Application to the Economy Car Selection Problem; Goicoechea, Burden, Sanchez; GMU [GOIC89RA]
5. Software Metrics: Their Measurement and Use in Models; Ambrose Goicoechea, GMU [GOIC89RC]

PORTLAND STATE UNIVERSITY

1. Tool Info. (PC-Metric - Developer's Toolbox)
2. Software Science and Weyuker's Fifth Property, Warren Harrison, PSU [HARR89RC]
3. A Note on the Berry-Meekings Style Metric; Warren Harrison & Curtis Cook; Communications of the ACM, 2/86 [HARR8602]
4. MAE: A Syntactic Metric Analysis Environment; Warren Harrison, Journal of Systems and Software, 1988 [HARR88]
5. Applying McCabe's Complexity Measure to Multiple-Exit Programs; Harrison, PSU [HARR89RA]
6. Using Software Metrics to Allocate Testing Resources; Harrison; Journal of Management Information Systems, Spring 88 [HARR8804]
7. Applying Software Complexity Metrics to Program Maintenance; Harrison, Magel, Kluczny, DeKock; Computer Magazine, 9/82 [HARR8209]

8. A Complexity Measure Based on Nesting Level; Harrison & Magel; Univ. of Mo. [HARR89RB]

9. Are Deeply Nested Conditionals Less Readable?; Harrison & Cook; Journal of Systems & Software, 1986 [HARR86]

10. A Micro/Macro Measure of Software Complexity; Harrison & Cook; Journal of Systems & Software, 1987 [HARR87]

TOOLS, COURSES AND NEWSLETTERS

1. Automated Systems Department (TBE-HSV) S/W Tools
2. TBE Accomplishments & Initiatives in Productivity, Training & Automated IV&V Tools - April 1988
3. IV&V Methodology & Tools Update - Computer Applications Directorate - April 1988
4. Timing & Sizing Simulation Evaluation - DP-Sim & Network II.5 - Terry Morris - April 1988
5. TBE 1988 BAMD V&V Tools Matrix
6. Logiscope Descriptions
7. McCabe & Associate Tools
8. AT&T Tools for Measuring Software Reliability
9. Dynamics Research Corporation -- AdaMat & Ada Quality Quarterly Newsletter
10. EVB, Software Engineering courses for 1989
11. Ada Information Clearinghouse newletters

CONFERENCE INFORMATION

1. Proceedings of the Joint Ada Conference - Fifth National Conference on Ada Technology and Washington Ada Symposium; March 16-19, 1987
2. Agenda, NSIA Software Quality and Productivity; March 10-12, 1987
3. Agenda, 9th Annual Conference on Multiple-Criteria Decision Making; August 1990
4. Agenda; Ada Project Management Course by Don Firesmith; Spring 1989.

APPENDIX A

PROPOSED IEEE METRICS TERMINOLOGY

PROPOSED IEEE STANDARD QUALITY AND RELIABILITY METRICS TERMINOLOGY

Term/Phrase **Definition**

Concept Phase

The period of time in the software life cycle during which system concepts and objectives needed by the user are identified and documented. Precedes the Requirements Phase.

Critical Range

Metric values used to classify software into categories of acceptable, marginal and unacceptable.

Critical Value

Metric value of a validated metric which is used to identify software which has unacceptable quality.

Defect

A product anomaly. Examples include such things as: 1) omissions and imperfections found during early life cycle phases and 2) faults contained in software sufficiently mature for test or operation. See also "fault".

Direct Metric

A metric that represents and defines a software quality factor, and which is valid by definition (e.g., mean time to software failure for the factor reliability).

Error

Human action that results in software containing a fault. Examples include omission or misinterpretation of user requirements in a software specification, incorrect translation or omission of a requirement in the design specification (ANSI/IEEE Std 729-1983).

Factor Sample

A set of factor values which is drawn from the metrics data base and used in metrics validation.

Factor Value

A value (see "metric value") of the direct metric that represents a factor.

Failure

1) The termination of the ability of a functional unit to perform its required function (ISO; ANSI/IEEE Std 729-1983). 2) An event in which a system or system component does not perform a required function within specified limits. A failure may be produced when a fault is encountered (ANSI/IEEE Std 729-1983).

Fault

1) An accidental condition that causes a functional unit to fail to perform its required function (ISO, ANSI/IEEE Std 729-1983). 2) A manifestation of an error in software. A fault, if encountered, may cause a failure. Synonymous with bug (ANSI/IEEE Std 729-1983).

Measure

1) A quantitative assessment of the degree to which a software product or process possesses a given attribute (IEEE P982.1/D3.0). 2) To ascertain or appraise by comparing to a standard; to apply a metric (IEEE P1061).

Measurement

1) The act or process of measuring. 2) A figure, extent, or amount obtained by measuring.

Metrics Framework

A tool used for organizing, selecting, communicating and evaluating the required quality attributes for a software system; a hierarchical breakdown of factors, subfactors and metrics for a software system.

Metrics Sample

A set of metrics values which is drawn from the metrics data base and used in metrics validation.

Metric Validation

The act or process of ensuring that a metric correctly predicts or accesses a quality factor.

Metric Value

An element from the range of a metric; a metric output.

Predictive Assessment

The process of using a predictor metric(s) to predict the value of another metric.

Predictive Metric

A metric which is used to predict the values of another metric.

Primitive

Data relating to the development or use of software that is used in developing measures or quantitative descriptions of software. Primitives are directly measurable or countable, or may be given a constant value or condition for a specific measure. Examples include error, failure, fault, time, time interval, date, number of noncommentary source code statements, edges, nodes.

Process Step

Any task performed in the development, implementation or maintenance of software (e.g., identify the software components of a system as part of the design).

Process Metric

Metrics used to measure characteristics of the methods, techniques, and tools employed in acquiring, developing, verifying, operating and changing the software system.

Product Metric

Metrics used to measure the characteristics of the documentation and code.

Quality Attribute

A characteristic of software; a generic term applying to factors, sub-factors, or metric values.

Quality Factor

An attribute of software that contributes to its quality.

Quality Requirement

A requirement that a software attribute be present in software to satisfy a contract, standard, specification, or other formally imposed.

Quality Sub-factor

A decomposition of a quality factor or quality sub-factor document.

Sample Software

Software selected from a current or completed project from which data can be obtained for use in preliminary testing of data collection and metric computation procedures.

Software Component

General term used to refer to an element of a software system, such as module, unit, data or document.

Software Quality Metric

A function whose inputs are software data and whose output is a single (numerical) value that can be interpreted as the degree to which software possesses a given attribute that affects its quality.

Software Reliability

The probability that software will not cause the failure of a system for a specified time under specified conditions. The probability is a function of the inputs to and use of the system as well as a function of the existence of faults in the software. The inputs to the system determined whether existing faults, if any, are encountered (ANSI/IEEE Std 729-1983).

Software Reliability Management

The process of optimizing the reliability of software through a program which emphasizes software error prevention, fault detection and removal, and the use of measurements to maximize reliability in light of project constraints such as resources (cost), schedule, and performance.

Validated Metric

A metric whose values have been statistically associated with corresponding quality factor values.

APPENDIX B

METRICS RANKING TABLES BY SUBFUNCTION

< B-1 through B-36 >

Table B-1 Subfunction Ranked Attributes & Metric Models

Subfunction Title: Detect Plumes

	SUBFUNCTION ATTRIBUTES, RANKINGS & SCORES								
METRIC CLASSES & MODELS	Reli	Surv	Intg	Thru	Usab	Main	Port	Reus	Effc
	H	H	H	H	L	L	L	L	M
Time Between Failure	5	5	5	5	1	1	1	1	3
Jelinski/Moranda	5	5	0	0	0	0	0	0	0
Schick/Wolverton (Linear)	5	5	0	0	0	0	0	0	0
Schick/Wolverton (Parabolic)	5	5	0	0	0	0	0	0	0
Moranda (Geometric De-eut)	5	0	0	0	0	0	0	0	0
Moranda (Hybrid Geomet Poiss)	5	0	0	0	0	0	0	0	0
Goel/Okumoto	5	0	0	0	0	0	0	0	0
Littlewood/Verrall	5	0	0	0	0	0	0	0	0
Lloyd/Lipow	5	0	0	0	0	0	0	0	0
Complexity									
Halstead	5	0	0	0	0	1	0	0	0
McCabe	5	0	0	0	0	1	0	0	0
Woodward (Knot Counts)	5	0	0	0	0	1	0	0	0
Chen (Nested Decision Stmts)	5	0	0	0	0	1	0	0	0
Gaffney	5	0	0	0	0	1	0	0	0
Benyon-Tinker	5	0	0	0	0	1	0	0	0
Gilb's (Binary Decision)	5	0	0	0	0	1	0	0	0
Chapin's Q	5	0	0	0	0	1	0	0	0
Segment-Global Usage Pair	5	0	0	0	0	1	0	0	0
Myer's (McCabe extension)	5	0	0	0	0	1	0	0	0
Hansen's (McCabe/Halstead)	5	0	0	0	0	1	0	0	0
Oviedo's (Data/Ctrl Flows)	5	0	0	0	0	1	0	0	0
Failure Count									
Geol/Okumoto	5	0	0	0	0	0	0	0	0
Schneidewind	5	0	0	0	0	0	0	0	0
Geol	5	0	0	0	0	0	0	0	0
Musa	5	0	0	0	0	0	0	0	0
Shooman	5	0	0	0	0	0	0	0	0
Moranda	5	5	0	0	0	0	0	0	0
Jelinski/Moranda	5	5	0	0	0	0	0	0	0
Moranda	5	5	0	0	0	0	0	0	0
Schick/Wolverton	5	5	0	0	0	0	0	0	0
Goel/Okumoto (Gen Poisson)	5	5	0	0	0	0	0	0	0
Brooks/Motley	5	5	0	0	0	0	0	0	0
IBM Poisson	5	5	0	0	0	0	0	0	0
Fault Seeding									
Mills	5	0	0	0	0	0	0	0	0
Input Domain Based									
Nelson	5	0	0	0	0	0	0	0	0
Ho	5	0	0	0	0	0	0	0	0
Ramamoorthy/Bastani	5	0	0	0	0	0	0	0	0

Table B-2 Subfunction Ranked Attributes & Metric Models

Subfunction Title: Detect Cold Bodies

METRIC CLASSES & MODELS	SUBFUNCTION ATTRIBUTES, RANKINGS & SCORES								
	Reli	Surv	Intg	Thru	Usab	Main	Port	Reus	Effc
	H	H	H	H	L	L	L	L	H
Time Between Failure	5	5	5	5	1	1	1	1	5
Jelinski/Moranda	5	5	0	0	0	0	0	0	0
Schick/Wolverton (Linear)	5	5	0	0	0	0	0	0	0
Schick/Wolverton (Parabolic)	5	5	0	0	0	0	0	0	0
Moranda (Geometric De-eut)	5	0	0	0	0	0	0	0	0
Moranda (Hybrid Geomet Poiss)	5	0	0	0	0	0	0	0	0
Goel/Okumoto	5	0	0	0	0	0	0	0	0
Littlewood/Verrall	5	0	0	0	0	0	0	0	0
Lloyd/Lipow	5	0	0	0	0	0	0	0	0
Complexity									
Halstead	5	0	0	0	0	1	0	0	0
McCabe	5	0	0	0	0	1	0	0	0
Woodward (Knot Counts)	5	0	0	0	0	1	0	0	0
Chen (Nested Decision Stmts)	5	0	0	0	0	1	0	0	0
Gaffney	5	0	0	0	0	1	0	0	0
Benyon-Tinker	5	0	0	0	0	1	0	0	0
Gilb's (Binary Decision)	5	0	0	0	0	1	0	0	0
Chapin's Q	5	0	0	0	0	1	0	0	0
Segment-Global Usage Pair	5	0	0	0	0	1	0	0	0
Myer's (McCabe extension)	5	0	0	0	0	1	0	0	0
Hansen's (McCabe/Halstead)	5	0	0	0	0	1	0	0	0
Oviedo's (Data/Ctrl Flows)	5	0	0	0	0	1	0	0	0
Failure Count									
Geol/Okumoto	5	0	0	0	0	0	0	0	0
Schneidewind	5	0	0	0	0	0	0	0	0
Geol	5	0	0	0	0	0	0	0	0
Musa	5	0	0	0	0	0	0	0	0
Shooman	5	0	0	0	0	0	0	0	0
Moranda	5	5	0	0	0	0	0	0	0
Jelinski/Moranda	5	5	0	0	0	0	0	0	0
Moranda	5	5	0	0	0	0	0	0	0
Schick/Wolverton	5	5	0	0	0	0	0	0	0
Goel/Okumoto (Gen Poisson)	5	5	0	0	0	0	0	0	0
Brooks/Motley	5	5	0	0	0	0	0	0	0
IBM Poisson	5	5	0	0	0	0	0	0	0
Fault Seeding									
Mills	5	0	0	0	0	0	0	0	0
Input Domain Based									
Nelson	5	0	0	0	0	0	0	0	0
Ho	5	0	0	0	0	0	0	0	0
Ramamoorthy/Bastani	5	0	0	0	0	0	0	0	0

Table B-3 Subfunction Ranked Attributes & Metric Models

Subfunction Title: RF Detect

METRIC CLASSES & MODELS	SUBFUNCTION ATTRIBUTES, RANKINGS & SCORES								
	Reli	Surv	Intg	Thru	Usab	Main	Port	Reus	Effc
	H	H	H	H	L	M	M	L	M
Time Between Failure	5	5	5	5	1	3	3	1	3
Jelinski/Moranda	5	5	0	0	0	0	0	0	0
Schick/Wolverton (Linear)	5	5	0	0	0	0	0	0	0
Schick/Wolverton (Parabolic)	5	5	0	0	0	0	0	0	0
Moranda (Geometric De-eut)	5	0	0	0	0	0	0	0	0
Moranda (Hybrid Geomet Poiss)	5	0	0	0	0	0	0	0	0
Goel/Okumoto	5	0	0	0	0	0	0	0	0
Littlewood/Verrall	5	0	0	0	0	0	0	0	0
Lloyd/Lipow	5	0	0	0	0	0	0	0	0
Complexity									
Halstead	5	0	0	0	0	3	0	0	0
McCabe	5	0	0	0	0	3	0	0	0
Woodward (Knot Counts)	5	0	0	0	0	3	0	0	0
Chen (Nested Decision Stmts)	5	0	0	0	0	3	0	0	0
Gaffney	5	0	0	0	0	3	0	0	0
Benyon-Tinker	5	0	0	0	0	3	0	0	0
Gilb's (Binary Decision)	5	0	0	0	0	3	0	0	0
Chapin's Q	5	0	0	0	0	3	0	0	0
Segment-Global Usage Pair	5	0	0	0	0	3	0	0	0
Myer's (McCabe extension)	5	0	0	0	0	3	0	0	0
Hansen's (McCabe/Halstead)	5	0	0	0	0	3	0	0	0
Oviedo's (Data/Ctrl Flows)	5	0	0	0	0	3	0	0	0
Failure Count									
Geol/Okumoto	5	0	0	0	0	0	0	0	0
Schneidewind	5	0	0	0	0	0	0	0	0
Geol	5	0	0	0	0	0	0	0	0
Musa	5	0	0	0	0	0	0	0	0
Shooman	5	0	0	0	0	0	0	0	0
Moranda	5	5	0	0	0	0	0	0	0
Jelinski/Moranda	5	5	0	0	0	0	0	0	0
Moranda	5	5	0	0	0	0	0	0	0
Schick/Wolverton	5	5	0	0	0	0	0	0	0
Goel/Okumoto (Gen Poisson)	5	5	0	0	0	0	0	0	0
Brooks/Motley	5	5	0	0	0	0	0	0	0
IBM Poisson	5	5	0	0	0	0	0	0	0
Fault Seeding									
Mills	5	0	0	0	0	0	0	0	0
Input Domain Based									
Nelson	5	0	0	0	0	0	0	0	0
Ho	5	0	0	0	0	0	0	0	0
Ramamoorthy/Bastani	5	0	0	0	0	0	0	0	0

Table B-4 Subfunction Ranked Attributes & Metric Models

Subfunction Title: Resolve Objects

METRIC CLASSES & MODELS	SUBFUNCTION ATTRIBUTES, RANKINGS & SCORES								
	Reli	Surv	Intg	Thru	Usab	Main	Port	Reus	Effc
	H	H	H	M	L	L	L	L	M
Time Between Failure	5	5	5	3	1	1	1	1	3
Jelinski/Moranda	5	5	0	0	0	0	0	0	0
Schick/Wolverton (Linear)	5	5	0	0	0	0	0	0	0
Schick/Wolverton (Parabolic)	5	5	0	0	0	0	0	0	0
Moranda (Geometric De-eut)	5	0	0	0	0	0	0	0	0
Moranda (Hybrid Geomet Poiss)	5	0	0	0	0	0	0	0	0
Goel/Okumoto	5	0	0	0	0	0	0	0	0
Littlewood/Verrall	5	0	0	0	0	0	0	0	0
Lloyd/Lipow	5	0	0	0	0	0	0	0	0
Complexity									
Halstead	5	0	0	0	0	1	0	0	0
McCabe	5	0	0	0	0	1	0	0	0
Woodward (Knot Counts)	5	0	0	0	0	1	0	0	0
Chen (Nested Decision Stmts)	5	0	0	0	0	1	0	0	0
Gaffney	5	0	0	0	0	1	0	0	0
Benyon-Tinker	5	0	0	0	0	1	0	0	0
Gilb's (Binary Decision)	5	0	0	0	0	1	0	0	0
Chapin's Q	5	0	0	0	0	1	0	0	0
Segment-Global Usage Pair	5	0	0	0	0	1	0	0	0
Myer's (McCabe extension)	5	0	0	0	0	1	0	0	0
Hansen's (McCabe/Halstead)	5	0	0	0	0	1	0	0	0
Oviedo's (Data/Ctrl Flows)	5	0	0	0	0	1	0	0	0
Failure Count									
Geol/Okumoto	5	0	0	0	0	0	0	0	0
Schneidewind	5	0	0	0	0	0	0	0	0
Geol	5	0	0	0	0	0	0	0	0
Musa	5	0	0	0	0	0	0	0	0
Shooman	5	0	0	0	0	0	0	0	0
Moranda	5	5	0	0	0	0	0	0	0
Jelinski/Moranda	5	5	0	0	0	0	0	0	0
Moranda	5	5	0	0	0	0	0	0	0
Schick/Wolverton	5	5	0	0	0	0	0	0	0
Goel/Okumoto (Gen Poisson)	5	5	0	0	0	0	0	0	0
Brooks/Motley	5	5	0	0	0	0	0	0	0
IBM Poisson	5	5	0	0	0	0	0	0	0
Fault Seeding									
Mills	5	0	0	0	0	0	0	0	0
Input Domain Based									
Nelson	5	0	0	0	0	0	0	0	0
Ho	5	0	0	0	0	0	0	0	0
Ramamoorthy/Bastani	5	0	0	0	0	0	0	0	0

Table B-5 Subfunction Ranked Attributes & Metric Models

Subfunction Title: Discriminate

METRIC CLASSES & MODELS	SUBFUNCTION ATTRIBUTES, RANKINGS & SCORES								
	Reli	Surv	Intg	Thru	Usab	Main	Port	Reus	Effc
	H	H	H	M	L	L	L	L	M
Time Between Failure	5	5	5	3	1	1	1	1	3
Jelinski/Moranda	5	5	0	0	0	0	0	0	0
Schick/Wolverton (Linear)	5	5	0	0	0	0	0	0	0
Schick/Wolverton (Parabolic)	5	5	0	0	0	0	0	0	0
Moranda (Geometric De-eut)	5	0	0	0	0	0	0	0	0
Moranda (Hybrid Geomet Poiss)	5	0	0	0	0	0	0	0	0
Goel/Okumoto	5	0	0	0	0	0	0	0	0
Littlewood/Verrall	5	0	0	0	0	0	0	0	0
Lloyd/Lipow	5	0	0	0	0	0	0	0	0
Complexity									
Halstead	5	0	0	0	0	1	0	0	0
McCabe	5	0	0	0	0	1	0	0	0
Woodward (Knot Counts)	5	0	0	0	0	1	0	0	0
Chen (Nested Decision Stmts)	5	0	0	0	0	1	0	0	0
Gaffney	5	0	0	0	0	1	0	0	0
Benyon-Tinker	5	0	0	0	0	1	0	0	0
Gilb's (Binary Decision)	5	0	0	0	0	1	0	0	0
Chapin's Q	5	0	0	0	0	1	0	0	0
Segment-Global Usage Pair	5	0	0	0	0	1	0	0	0
Myer's (McCabe extension)	5	0	0	0	0	1	0	0	0
Hansen's (McCabe/Halstead)	5	0	0	0	0	1	0	0	0
Oviedo's (Data/Ctrl Flows)	5	0	0	0	0	1	0	0	0
Failure Count									
Geol/Okumoto	5	0	0	0	0	0	0	0	0
Schneidewind	5	0	0	0	0	0	0	0	0
Geol	5	0	0	0	0	0	0	0	0
Musa	5	0	0	0	0	0	0	0	0
Shooman	5	0	0	0	0	0	0	0	0
Moranda	5	5	0	0	0	0	0	0	0
Jelinski/Moranda	5	5	0	0	0	0	0	0	0
Moranda	5	5	0	0	0	0	0	0	0
Schick/Wolverton	5	5	0	0	0	0	0	0	0
Goel/Okumoto (Gen Poisson)	5	5	0	0	0	0	0	0	0
Brooks/Motley	5	5	0	0	0	0	0	0	0
IBM Poisson	5	5	0	0	0	0	0	0	0
Fault Seeding									
Mills	5	0	0	0	0	0	0	0	0
Input Domain Based									
Nelson	5	0	0	0	0	0	0	0	0
Ho	5	0	0	0	0	0	0	0	0
Ramamoorthy/Bastani	5	0	0	0	0	0	0	0	0

Table B-6 Subfunction Ranked Attributes & Metric Models

Subfunction Title: Assess Kills

METRIC CLASSES & MODELS	SUBFUNCTION ATTRIBUTES, RANKINGS & SCORES								
	Reli	Surv	Intg	Thru	Usab	Main	Port	Reus	Effc
	H	H	H	M	M	M	M	L	M
Time Between Failure	5	5	5	3	3	3	3	1	3
Jelinski/Moranda	5	5	0	0	0	0	0	0	0
Schick/Wolverton (Linear)	5	5	0	0	0	0	0	0	0
Schick/Wolverton (Parabolic)	5	5	0	0	0	0	0	0	0
Moranda (Geometric De-eut)	5	0	0	0	0	0	0	0	0
Moranda (Hybrid Geomet Poiss)	5	0	0	0	0	0	0	0	0
Goel/Okumoto	5	0	0	0	0	0	0	0	0
Littlewood/Verrall	5	0	0	0	0	0	0	0	0
Lloyd/Lipow	5	0	0	0	0	0	0	0	0
Complexity									
Halstead	5	0	0	0	0	3	0	0	0
McCabe	5	0	0	0	0	3	0	0	0
Woodward (Knot Counts)	5	0	0	0	0	3	0	0	0
Chen (Nested Decision Stmts)	5	0	0	0	0	3	0	0	0
Gaffney	5	0	0	0	0	3	0	0	0
Benyon-Tinker	5	0	0	0	0	3	0	0	0
Gilb's (Binary Decision)	5	0	0	0	0	3	0	0	0
Chapin's Q	5	0	0	0	0	3	0	0	0
Segment-Global Usage Pair	5	0	0	0	0	3	0	0	0
Myer's (McCabe extension)	5	0	0	0	0	3	0	0	0
Hansen's (McCabe/Halstead)	5	0	0	0	0	3	0	0	0
Oviedo's (Data/Ctrl Flows)	5	0	0	0	0	3	0	0	0
Failure Count									
Geol/Okumoto	5	0	0	0	0	0	0	0	0
Schneidewind	5	0	0	0	0	0	0	0	0
Geol	5	0	0	0	0	0	0	0	0
Musa	5	0	0	0	0	0	0	0	0
Shooman	5	0	0	0	0	0	0	0	0
Moranda	5	5	0	0	0	0	0	0	0
Jelinski/Moranda	5	5	0	0	0	0	0	0	0
Moranda	5	5	0	0	0	0	0	0	0
Schick/Wolverton	5	5	0	0	0	0	0	0	0
Goel/Okumoto (Gen Poisson)	5	5	0	0	0	0	0	0	0
Brooks/Motley	5	5	0	0	0	0	0	0	0
IBM Poisson	5	5	0	0	0	0	0	0	0
Fault Seeding									
Mills	5	0	0	0	0	0	0	0	0
Input Domain Based									
Nelson	5	0	0	0	0	0	0	0	0
Ho	5	0	0	0	0	0	0	0	0
Ramamoorthy/Bastani	5	0	0	0	0	0	0	0	0

Table B-7 Subfunction Ranked Attributes & Metric Models

Subfunction Title: Correlate

METRIC CLASSES & MODELS	SUBFUNCTION ATTRIBUTES, RANKINGS & SCORES								
	Reli	Surv	Intg	Thru	Usab	Main	Port	Reus	Effc
	H	H	H	H	L	L	L	L	M
Time Between Failure	5	5	5	5	1	1	1	1	3
Jelinski/Moranda	5	5	0	0	0	0	0	0	0
Schick/Wolverton (Linear)	5	5	0	0	0	0	0	0	0
Schick/Wolverton (Parabolic)	5	5	0	0	0	0	0	0	0
Moranda (Geometric De-eut)	5	0	0	0	0	0	0	0	0
Moranda (Hybrid Geomet Poiss)	5	0	0	0	0	0	0	0	0
Goel/Okumoto	5	0	0	0	0	0	0	0	0
Littlewood/Verrall	5	0	0	0	0	0	0	0	0
Lloyd/Lipow	5	0	0	0	0	0	0	0	0
Complexity									
Halstead	5	0	0	0	0	1	0	0	0
McCabe	5	0	0	0	0	1	0	0	0
Woodward (Knot Counts)	5	0	0	0	0	1	0	0	0
Chen (Nested Decision Stmts)	5	0	0	0	0	1	0	0	0
Gaffney	5	0	0	0	0	1	0	0	0
Benyon-Tinker	5	0	0	0	0	1	0	0	0
Gilb's (Binary Decision)	5	0	0	0	0	1	0	0	0
Chapin's Q	5	0	0	0	0	1	0	0	0
Segment-Global Usage Pair	5	0	0	0	0	1	0	0	0
Myer's (McCabe extension)	5	0	0	0	0	1	0	0	0
Hansen's (McCabe/Halstead)	5	0	0	0	0	1	0	0	0
Oviedo's (Data/Ctrl Flows)	5	0	0	0	0	1	0	0	0
Failure Count									
Geol/Okumoto	5	0	0	0	0	0	0	0	0
Schneidewind	5	0	0	0	0	0	0	0	0
Geol	5	0	0	0	0	0	0	0	0
Musa	5	0	0	0	0	0	0	0	0
Shooman	5	0	0	0	0	0	0	0	0
Moranda	5	5	0	0	0	0	0	0	0
Jelinski/Moranda	5	5	0	0	0	0	0	0	0
Moranda	5	5	0	0	0	0	0	0	0
Schick/Wolverton	5	5	0	0	0	0	0	0	0
Goel/Okumoto (Gen Poisson)	5	5	0	0	0	0	0	0	0
Brooks/Motley	5	5	0	0	0	0	0	0	0
IBM Poisson	5	5	0	0	0	0	0	0	0
Fault Seeding									
Mills	5	0	0	0	0	0	0	0	0
Input Domain Based									
Nelson	5	0	0	0	0	0	0	0	0
Ho	5	0	0	0	0	0	0	0	0
Ramamoorthy/Bastani	5	0	0	0	0	0	0	0	0

Table B-8 Subfunction Ranked Attributes & Metric Models

Subfunction Title: Initiate Track

METRIC CLASSES & MODELS	SUBFUNCTION ATTRIBUTES, RANKINGS & SCORES								
	Reli	Surv	Intg	Thru	Usab	Main	Port	Reus	Effc
	H	H	H	H	L	L	L	L	M
Time Between Failure	5	5	5	5	1	1	1	1	3
Jelinski/Moranda	5	5	0	0	0	0	0	0	0
Schick/Wolverton (Linear)	5	5	0	0	0	0	0	0	0
Schick/Wolverton (Parabolic)	5	5	0	0	0	0	0	0	0
Moranda (Geometric De-eut)	5	0	0	0	0	0	0	0	0
Moranda (Hybrid Geomet Poiss)	5	0	0	0	0	0	0	0	0
Goel/Okumoto	5	0	0	0	0	0	0	0	0
Littlewood/Verrall	5	0	0	0	0	0	0	0	0
Lloyd/Lipow	5	0	0	0	0	0	0	0	0
Complexity									
Halstead	5	0	0	0	0	1	0	0	0
McCabe	5	0	0	0	0	1	0	0	0
Woodward (Knot Counts)	5	0	0	0	0	1	0	0	0
Chen (Nested Decision Stmts)	5	0	0	0	0	1	0	0	0
Gaffney	5	0	0	0	0	1	0	0	0
Benyon-Tinker	5	0	0	0	0	1	0	0	0
Gilb's (Binary Decision)	5	0	0	0	0	1	0	0	0
Chapin's Q	5	0	0	0	0	1	0	0	0
Segment-Global Usage Pair	5	0	0	0	0	1	0	0	0
Myer's (McCabe extension)	5	0	0	0	0	1	0	0	0
Hansen's (McCabe/Halstead)	5	0	0	0	0	1	0	0	0
Oviedo's (Data/Ctrl Flows)	5	0	0	0	0	1	0	0	0
Failure Count									
Geol/Okumoto	5	0	0	0	0	0	0	0	0
Schneidewind	5	0	0	0	0	0	0	0	0
Geol	5	0	0	0	0	0	0	0	0
Musa	5	0	0	0	0	0	0	0	0
Shooman	5	0	0	0	0	0	0	0	0
Moranda	5	5	0	0	0	0	0	0	0
Jelinski/Moranda	5	5	0	0	0	0	0	0	0
Moranda	5	5	0	0	0	0	0	0	0
Schick/Wolverton	5	5	0	0	0	0	0	0	0
Goel/Okumoto (Gen Poisson)	5	5	0	0	0	0	0	0	0
Brooks/Motley	5	5	0	0	0	0	0	0	0
IBM Poisson	5	5	0	0	0	0	0	0	0
Fault Seeding									
Mills	5	0	0	0	0	1	0	0	0
Input Domain Based									
Nelson	5	0	0	0	0	0	0	0	0
Ho	5	0	0	0	0	0	0	0	0
Ramamoorthy/Bastani	5	0	0	0	0	0	0	0	0

Table B-9 Subfunction Ranked Attributes & Metric Models

Subfunction Title: Estimate State

METRIC CLASSES & MODELS	SUBFUNCTION ATTRIBUTES, RANKINGS & SCORES								
	Reli	Surv	Intg	Thru	Usab	Main	Port	Reus	Effc
	H	H	H	H	L	L	M	L	M
Time Between Failure	5	5	5	5	1	1	3	1	3
Jelinski/Moranda	5	5	0	0	0	0	0	0	0
Schick/Wolverton (Linear)	5	5	0	0	0	0	0	0	0
Schick/Wolverton (Parabolic)	5	5	0	0	0	0	0	0	0
Moranda (Geometric De-eut)	5	0	0	0	0	0	0	0	0
Moranda (Hybrid Geomet Poiss)	5	0	0	0	0	0	0	0	0
Goel/Okumoto	5	0	0	0	0	0	0	0	0
Littlewood/Verrall	5	0	0	0	0	0	0	0	0
Lloyd/Lipow	5	0	0	0	0	0	0	0	0
Complexity									
Halstead	5	0	0	0	0	1	0	0	0
McCabe	5	0	0	0	0	1	0	0	0
Woodward (Knot Counts)	5	0	0	0	0	1	0	0	0
Chen (Nested Decision Stmts)	5	0	0	0	0	1	0	0	0
Gaffney	5	0	0	0	0	1	0	0	0
Benyon-Tinker	5	0	0	0	0	1	0	0	0
Gilb's (Binary Decision)	5	0	0	0	0	1	0	0	0
Chapin's Q	5	0	0	0	0	1	0	0	0
Segment-Global Usage Pair	5	0	0	0	0	1	0	0	0
Myer's (McCabe extension)	5	0	0	0	0	1	0	0	0
Hansen's (McCabe/Halstead)	5	0	0	0	0	1	0	0	0
Oviedo's (Data/Ctrl Flows)	5	0	0	0	0	1	0	0	0
Failure Count									
Geol/Okumoto	5	0	0	0	0	0	0	0	0
Schneidewind	5	0	0	0	0	0	0	0	0
Geol	5	0	0	0	0	0	0	0	0
Musa	5	0	0	0	0	0	0	0	0
Shooman	5	0	0	0	0	0	0	0	0
Moranda	5	5	0	0	0	0	0	0	0
Jelinski/Moranda	5	5	0	0	0	0	0	0	0
Moranda	5	5	0	0	0	0	0	0	0
Schick/Wolverton	5	5	0	0	0	0	0	0	0
Goel/Okumoto (Gen Poisson)	5	5	0	0	0	0	0	0	0
Brooks/Motley	5	5	0	0	0	0	0	0	0
IBM Poisson	5	5	0	0	0	0	0	0	0
Fault Seeding									
Mills	5	0	0	0	0	0	0	0	0
Input Domain Based									
Nelson	5	0	0	0	0	0	0	0	0
Ho	5	0	0	0	0	0	0	0	0
Ramamoorthy/Bastani	5	0	0	0	0	0	0	0	0

Table B-10 Subfunction Ranked Attributes & Metric Models

Subfunction Title: Predict Intercept and Impact Points

METRIC CLASSES & MODELS	SUBFUNCTION ATTRIBUTES, RANKINGS & SCORES								
	Reli	Surv	Intg	Thru	Usab	Main	Port	Reus	Effc
	H	H	H	M	L	L	L	L	M
Time Between Failure	5	5	5	3	1	1	1	1	3
Jelinski/Moranda	5	5	0	0	0	0	0	0	0
Schick/Wolverton (Linear)	5	5	0	0	0	0	0	0	0
Schick/Wolverton (Parabolic)	5	5	0	0	0	0	0	0	0
Moranda (Geometric De-eut)	5	0	0	0	0	0	0	0	0
Moranda (Hybrid Geomet Poiss)	5	0	0	0	0	0	0	0	0
Goel/Okumoto	5	0	0	0	0	0	0	0	0
Littlewood/Verrall	5	0	0	0	0	0	0	0	0
Lloyd/Lipow	5	0	0	0	0	0	0	0	0
Complexity									
Halstead	5	0	0	0	0	1	0	0	0
McCabe	5	0	0	0	0	1	0	0	0
Woodward (Knot Counts)	5	0	0	0	0	1	0	0	0
Chen (Nested Decision Stmts)	5	0	0	0	0	1	0	0	0
Gaffney	5	0	0	0	0	1	0	0	0
Benyon-Tinker	5	0	0	0	0	1	0	0	0
Gilb's (Binary Decision)	5	0	0	0	0	1	0	0	0
Chapin's Q	5	0	0	0	0	1	0	0	0
Segment-Global Usage Pair	5	0	0	0	0	1	0	0	0
Myer's (McCabe extension)	5	0	0	0	0	1	0	0	0
Hansen's (McCabe/Halstead)	5	0	0	0	0	1	0	0	0
Oviedo's (Data/Ctrl Flows)	5	0	0	0	0	1	0	0	0
Failure Count									
Geol/Okumoto	5	0	0	0	0	0	0	0	0
Schneidewind	5	0	0	0	0	0	0	0	0
Geol	5	0	0	0	0	0	0	0	0
Musa	5	0	0	0	0	0	0	0	0
Shooman	5	0	0	0	0	0	0	0	0
Moranda	5	5	0	0	0	0	0	0	0
Jelinski/Moranda	5	5	0	0	0	0	0	0	0
Moranda	5	5	0	0	0	0	0	0	0
Schick/Wolverton	5	5	0	0	0	0	0	0	0
Goel/Okumoto (Gen Poisson)	5	5	0	0	0	0	0	0	0
Brooks/Motley	5	5	0	0	0	0	0	0	0
IBM Poisson	5	5	0	0	0	0	0	0	0
Fault Seeding									
Mills	5	0	0	0	0	0	0	0	0
Input Domain Based									
Nelson	5	0	0	0	0	0	0	0	0
Ho	5	0	0	0	0	0	0	0	0
Ramamoorthy/Bastani	5	0	0	0	0	0	0	0	0

Table B-11 Subfunction Ranked Attributes & Metric Models

Subfunction Title: Interplatform Data Communications

METRIC CLASSES & MODELS	SUBFUNCTION ATTRIBUTES, RANKINGS & SCORES								
	Reli	Surv	Intg	Thru	Usab	Main	Port	Reus	Effc
	H	H	H	H	L	L	L	M	H
Time Between Failure	5	5	5	5	1	1	1	3	5
Jelinski/Moranda	5	5	0	0	0	0	0	0	0
Schick/Wolverton (Linear)	5	5	0	0	0	0	0	0	0
Schick/Wolverton (Parabolic)	5	5	0	0	0	0	0	0	0
Moranda (Geometric De-eut)	5	0	0	0	0	0	0	0	0
Moranda (Hybrid Geomet Poiss)	5	0	0	0	0	0	0	0	0
Goel/Okumoto	5	0	0	0	0	0	0	0	0
Littlewood/Verrall	5	0	0	0	0	0	0	0	0
Lloyd/Lipow	5	0	0	0	0	0	0	0	0
Complexity									
Halstead	5	0	0	0	0	1	0	0	0
McCabe	5	0	0	0	0	1	0	0	0
Woodward (Knot Counts)	5	0	0	0	0	1	0	0	0
Chen (Nested Decision Stmts)	5	0	0	0	0	1	0	0	0
Gaffney	5	0	0	0	0	1	0	0	0
Benyon-Tinker	5	0	0	0	0	1	0	0	0
Gilb's (Binary Decision)	5	0	0	0	0	1	0	0	0
Chapin's Q	5	0	0	0	0	1	0	0	0
Segment-Global Usage Pair	5	0	0	0	0	1	0	0	0
Myer's (McCabe extension)	5	0	0	0	0	1	0	0	0
Hansen's (McCabe/Halstead)	5	0	0	0	0	1	0	0	0
Oviedo's (Data/Ctrl Flows)	5	0	0	0	0	1	0	0	0
Failure Count									
Geol/Okumoto	5	0	0	0	0	0	0	0	0
Schneidewind	5	0	0	0	0	0	0	0	0
Geol	5	0	0	0	0	0	0	0	0
Musa	5	0	0	0	0	0	0	0	0
Shooman	5	0	0	0	0	0	0	0	0
Moranda	5	5	0	0	0	0	0	0	0
Jelinski/Moranda	5	5	0	0	0	0	0	0	0
Moranda	5	5	0	0	0	0	0	0	0
Schick/Wolverton	5	5	0	0	0	0	0	0	0
Goel/Okumoto (Gen Poisson)	5	5	0	0	0	0	0	0	0
Brooks/Motley	5	5	0	0	0	0	0	0	0
IBM Poisson	5	5	0	0	0	0	0	0	0
Fault Seeding									
Mills	5	0	0	0	0	0	0	0	0
Input Domain Based									
Nelson	5	0	0	0	0	0	0	0	0
Ho	5	0	0	0	0	0	0	0	0
Ramamoorthy/Bastani	5	0	0	0	0	0	0	0	0

Table B-12 Subfunction Ranked Attributes & Metric Models

Subfunction Title: Ground - Space Communications

METRIC CLASSES & MODELS	SUBFUNCTION ATTRIBUTES, RANKINGS & SCORES								
	Reli	Surv	Intg	Thru	Usab	Main	Port	Reus	Effc
	H	H	H	M	L	M	M	L	M
Time Between Failure	5	5	5	3	1	3	3	1	3
Jelinski/Moranda	5	5	0	0	0	0	0	0	0
Schick/Wolverton (Linear)	5	5	0	0	0	0	0	0	0
Schick/Wolverton (Parabolic)	5	5	0	0	0	0	0	0	0
Moranda (Geometric De-eut)	5	0	0	0	0	0	0	0	0
Moranda (Hybrid Geomet Poiss)	5	0	0	0	0	0	0	0	0
Goel/Okumoto	5	0	0	0	0	0	0	0	0
Littlewood/Verrall	5	0	0	0	0	0	0	0	0
Lloyd/Lipow	5	0	0	0	0	0	0	0	0
Complexity									
Halstead	5	0	0	0	0	3	0	0	0
McCabe	5	0	0	0	0	3	0	0	0
Woodward (Knot Counts)	5	0	0	0	0	3	0	0	0
Chen (Nested Decision Stmts)	5	0	0	0	0	3	0	0	0
Gaffney	5	0	0	0	0	3	0	0	0
Benyon-Tinker	5	0	0	0	0	3	0	0	0
Gilb's (Binary Decision)	5	0	0	0	0	3	0	0	0
Chapin's Q	5	0	0	0	0	3	0	0	0
Segment-Global Usage Pair	5	0	0	0	0	3	0	0	0
Myer's (McCabe extension)	5	0	0	0	0	3	0	0	0
Hansen's (McCabe/Halstead)	5	0	0	0	0	3	0	0	0
Oviedo's (Data/Ctrl Flows)	5	0	0	0	0	3	0	0	0
Failure Count									
Geol/Okumoto	5	0	0	0	0	0	0	0	0
Schneidewind	5	0	0	0	0	0	0	0	0
Geol	5	0	0	0	0	0	0	0	0
Musa	5	0	0	0	0	0	0	0	0
Shooman	5	0	0	0	0	0	0	0	0
Moranda	5	5	0	0	0	0	0	0	0
Jelinski/Moranda	5	5	0	0	0	0	0	0	0
Moranda	5	5	0	0	0	0	0	0	0
Schick/Wolverton	5	5	0	0	0	0	0	0	0
Goel/Okumoto (Gen Poisson)	5	5	0	0	0	0	0	0	0
Brooks/Motley	5	5	0	0	0	0	0	0	0
IBM Poisson	5	5	0	0	0	0	0	0	0
Fault Seeding									
Mills	5	0	0	0	0	0	0	0	0
Input Domain Based									
Nelson	5	0	0	0	0	0	0	0	0
Ho	5	0	0	0	0	0	0	0	0
Ramamoorthy/Bastani	5	0	0	0	0	0	0	0	0

Table B-13 Subfunction Ranked Attributes & Metric Models

Subfunction Title: Ground Communications

METRIC CLASSES & MODELS	SUBFUNCTION ATTRIBUTES, RANKINGS & SCORES								
	Reli	Surv	Intg	Thru	Usab	Main	Port	Reus	Effc
	M	M	M	M	M	H	M	H	L
Time Between Failure	3	3	3	3	3	5	3	5	1
Jelinski/Moranda	3	3	0	0	0	0	0	0	0
Schick/Wolverton (Linear)	3	3	0	0	0	0	0	0	0
Schick/Wolverton (Parabolic)	3	3	0	0	0	0	0	0	0
Moranda (Geometric De-eut)	3	0	0	0	0	0	0	0	0
Moranda (Hybrid Geomet Poiss)	3	0	0	0	0	0	0	0	0
Goel/Okumoto	3	0	0	0	0	0	0	0	0
Littlewood/Verrall	3	0	0	0	0	0	0	0	0
Lloyd/Lipow	3	0	0	0	0	0	0	0	0
Complexity									
Halstead	3	0	0	0	0	5	0	0	0
McCabe	3	0	0	0	0	5	0	0	0
Woodward (Knot Counts)	3	0	0	0	0	5	0	0	0
Chen (Nested Decision Stmts)	3	0	0	0	0	5	0	0	0
Gaffney	3	0	0	0	0	5	0	0	0
Benyon-Tinker	3	0	0	0	0	5	0	0	0
Gilb's (Binary Decision)	3	0	0	0	0	5	0	0	0
Chapin's Q	3	0	0	0	0	5	0	0	0
Segment-Global Usage Pair	3	0	0	0	0	5	0	0	0
Myer's (McCabe extension)	3	0	0	0	0	5	0	0	0
Hansen's (McCabe/Halstead)	3	0	0	0	0	5	0	0	0
Oviedo's (Data/Ctrl Flows)	3	0	0	0	0	5	0	0	0
Failure Count									
Geol/Okumoto	3	0	0	0	0	0	0	0	0
Schneidewind	3	0	0	0	0	0	0	0	0
Geol	3	0	0	0	0	0	0	0	0
Musa	3	0	0	0	0	0	0	0	0
Shooman	3	0	0	0	0	0	0	0	0
Moranda	3	3	0	0	0	0	0	0	0
Jelinski/Moranda	3	3	0	0	0	0	0	0	0
Moranda	3	3	0	0	0	0	0	0	0
Schick/Wolverton	3	3	0	0	0	0	0	0	0
Goel/Okumoto (Gen Poisson)	3	3	0	0	0	0	0	0	0
Brooks/Motley	3	3	0	0	0	0	0	0	0
IBM Poisson	3	3	0	0	0	0	0	0	0
Fault Seeding									
Mills	3	0	0	0	0	0	0	0	0
Input Domain Based									
Nelson	3	0	0	0	0	0	0	0	0
Ho	3	0	0	0	0	0	0	0	0
Ramamoorthy/Bastani	3	0	0	0	0	0	0	0	0

Table B-16 Subfunction Ranked Attributes & Metric Models

Subfunction Title: Assign and Control SBI Weapons

METRIC CLASSES & MODELS	SUBFUNCTION ATTRIBUTES, RANKINGS & SCORES								
	Reli	Surv	Intg	Thru	Usab	Main	Port	Reus	Effc
	H	H	H	H	L	L	L	L	M
Time Between Failure	5	5	5	5	1	1	1	1	3
Jelinski/Moranda	5	5	0	0	0	0	0	0	0
Schick/Wolverton (Linear)	5	5	0	0	0	0	0	0	0
Schick/Wolverton (Parabolic)	5	5	0	0	0	0	0	0	0
Moranda (Geometric De-eut)	5	0	0	0	0	0	0	0	0
Moranda (Hybrid Geomet Poiss)	5	0	0	0	0	0	0	0	0
Goel/Okumoto	5	0	0	0	0	0	0	0	0
Littlewood/Verrall	5	0	0	0	0	0	0	0	0
Lloyd/Lipow	5	0	0	0	0	0	0	0	0
Complexity									
Halstead	5	0	0	0	0	1	0	0	0
McCabe	5	0	0	0	0	1	0	0	0
Woodward (Knot Counts)	5	0	0	0	0	1	0	0	0
Chen (Nested Decision Stmts)	5	0	0	0	0	1	0	0	0
Gaffney	5	0	0	0	0	1	0	0	0
Benyon-Tinker	5	0	0	0	0	1	0	0	0
Gilb's (Binary Decision)	5	0	0	0	0	1	0	0	0
Chapin's Q	5	0	0	0	0	1	0	0	0
Segment-Global Usage Pair	5	0	0	0	0	1	0	0	0
Myer's (McCabe extension)	5	0	0	0	0	1	0	0	0
Hansen's (McCabe/Halstead)	5	0	0	0	0	1	0	0	0
Oviedo's (Data/Ctrl Flows)	5	0	0	0	0	1	0	0	0
Failure Count									
Geol/Okumoto	5	0	0	0	0	0	0	0	0
Schneidewind	5	0	0	0	0	0	0	0	0
Geol	5	0	0	0	0	0	0	0	0
Musa	5	0	0	0	0	0	0	0	0
Shooman	5	0	0	0	0	0	0	0	0
Moranda	5	5	0	0	0	0	0	0	0
Jelinski/Moranda	5	5	0	0	0	0	0	0	0
Moranda	5	5	0	0	0	0	0	0	0
Schick/Wolverton	5	5	0	0	0	0	0	0	0
Goel/Okumoto (Gen Poisson)	5	5	0	0	0	0	0	0	0
Brooks/Motley	5	5	0	0	0	0	0	0	0
IBM Poisson	5	5	0	0	0	0	0	0	0
Fault Seeding									
Mills	5	0	0	0	0	0	0	0	0
Input Domain Based									
Nelson	5	0	0	0	0	0	0	0	0
Ho	5	0	0	0	0	0	0	0	0
Ramamoorthy/Bastani	5	0	0	0	0	0	0	0	0

Table B-17 Subfunction Ranked Attributes & Metric Models

Subfunction Title: Assign and Control GBI Weapons

METRIC CLASSES & MODELS	SUBFUNCTION ATTRIBUTES, RANKINGS & SCORES								
	Reli	Surv	Intg	Thru	Usab	Main	Port	Reus	Effc
	H	H	H	M	L	H	L	L	M
Time Between Failure	5	5	5	3	1	5	1	1	3
Jelinski/Moranda	5	5	0	0	0	0	0	0	0
Schick/Wolverton (Linear)	5	5	0	0	0	0	0	0	0
Schick/Wolverton (Parabolic)	5	5	0	0	0	0	0	0	0
Moranda (Geometric De-eut)	5	0	0	0	0	0	0	0	0
Moranda (Hybrid Geomet Poiss)	5	0	0	0	0	0	0	0	0
Goel/Okumoto	5	0	0	0	0	0	0	0	0
Littlewood/Verrall	5	0	0	0	0	0	0	0	0
Lloyd/Lipow	5	0	0	0	0	0	0	0	0
Complexity									
Halstead	5	0	0	0	0	5	0	0	0
McCabe	5	0	0	0	0	5	0	0	0
Woodward (Knot Counts)	5	0	0	0	0	5	0	0	0
Chen (Nested Decision Stmts)	5	0	0	0	0	5	0	0	0
Gaffney	5	0	0	0	0	5	0	0	0
Benyon-Tinker	5	0	0	0	0	5	0	0	0
Gilb's (Binary Decision)	5	0	0	0	0	5	0	0	0
Chapin's Q	5	0	0	0	0	5	0	0	0
Segment-Global Usage Pair	5	0	0	0	0	5	0	0	0
Myer's (McCabe extension)	5	0	0	0	0	5	0	0	0
Hansen's (McCabe/Halstead)	5	0	0	0	0	5	0	0	0
Oviedo's (Data/Ctrl Flows)	5	0	0	0	0	5	0	0	0
Failure Count									
Geol/Okumoto	5	0	0	0	0	0	0	0	0
Schneidewind	5	0	0	0	0	0	0	0	0
Geol	5	0	0	0	0	0	0	0	0
Musa	5	0	0	0	0	0	0	0	0
Shooman	5	0	0	0	0	0	0	0	0
Moranda	5	5	0	0	0	0	0	0	0
Jelinski/Moranda	5	5	0	0	0	0	0	0	0
Moranda	5	5	0	0	0	0	0	0	0
Schick/Wolverton	5	5	0	0	0	0	0	0	0
Goel/Okumoto (Gen Poisson)	5	5	0	0	0	0	0	0	0
Brooks/Motley	5	5	0	0	0	0	0	0	0
IBM Poisson	5	5	0	0	0	0	0	0	0
Fault Seeding									
Mills	5	0	0	0	0	0	0	0	0
Input Domain Based									
Nelson	5	0	0	0	0	0	0	0	0
Ho	5	0	0	0	0	0	0	0	0
Ramamoorthy/Bastani	5	0	0	0	0	0	0	0	0

Table B-18 Subfunction Ranked Attributes & Metric Models

Subfunction Title: Guide and Control SBI Weapons

METRIC CLASSES & MODELS	SUBFUNCTION ATTRIBUTES, RANKINGS & SCORES								
	Reli	Surv	Intg	Thru	Usab	Main	Port	Reus	Effc
	H	H	H	H	L	L	L	L	H
Time Between Failure	5	5	5	5	1	1	1	1	5
Jelinski/Moranda	5	5	0	0	0	0	0	0	0
Schick/Wolverton (Linear)	5	5	0	0	0	0	0	0	0
Schick/Wolverton (Parabolic)	5	5	0	0	0	0	0	0	0
Moranda (Geometric De-eut)	5	0	0	0	0	0	0	0	0
Moranda (Hybrid Geomet Poiss)	5	0	0	0	0	0	0	0	0
Goel/Okumoto	5	0	0	0	0	0	0	0	0
Littlewood/Verrall	5	0	0	0	0	0	0	0	0
Lloyd/Lipow	5	0	0	0	0	0	0	0	0
Complexity									
Halstead	5	0	0	0	0	1	0	0	0
McCabe	5	0	0	0	0	1	0	0	0
Woodward (Knot Counts)	5	0	0	0	0	1	0	0	0
Chen (Nested Decision Stmts)	5	0	0	0	0	1	0	0	0
Gaffney	5	0	0	0	0	1	0	0	0
Benyon-Tinker	5	0	0	0	0	1	0	0	0
Gilb's (Binary Decision)	5	0	0	0	0	1	0	0	0
Chapin's Q	5	0	0	0	0	1	0	0	0
Segment-Global Usage Pair	5	0	0	0	0	1	0	0	0
Myer's (McCabe extension)	5	0	0	0	0	1	0	0	0
Hansen's (McCabe/Halstead)	5	0	0	0	0	1	0	0	0
Oviedo's (Data/Ctrl Flows)	5	0	0	0	0	1	0	0	0
Failure Count									
Geol/Okumoto	5	0	0	0	0	0	0	0	0
Schneidewind	5	0	0	0	0	0	0	0	0
Geol	5	0	0	0	0	0	0	0	0
Musa	5	0	0	0	0	0	0	0	0
Shooman	5	0	0	0	0	0	0	0	0
Moranda	5	5	0	0	0	0	0	0	0
Jelinski/Moranda	5	5	0	0	0	0	0	0	0
Moranda	5	5	0	0	0	0	0	0	0
Schick/Wolverton	5	5	0	0	0	0	0	0	0
Goel/Okumoto (Gen Poisson)	5	5	0	0	0	0	0	0	0
Brooks/Motley	5	5	0	0	0	0	0	0	0
IBM Poisson	5	5	0	0	0	0	0	[illegible]	0
Fault Seeding									
Mills	5	0	0	0	0	0	0	0	0
Input Domain Based									
Nelson	5	0	0	0	0	0	0	0	0
Ho	5	0	0	0	0	0	0	0	0
Ramamoorthy/Bastani	5	0	0	0	0	0	0	0	0

Table B-19 Subfunction Ranked Attributes & Metric Models

Subfunction Title: Guide and Control GBI Weapons

METRIC CLASSES & MODELS	SUBFUNCTION ATTRIBUTES, RANKINGS & SCORES								
	Reli	Surv	Intg	Thru	Usab	Main	Port	Reus	Effc
	H	H	H	M	L	H	L	L	M
Time Between Failure	5	5	5	3	1	5	1	1	3
Jelinski/Moranda	5	5	0	0	0	0	0	0	0
Schick/Wolverton (Linear)	5	5	0	0	0	0	0	0	0
Schick/Wolverton (Parabolic)	5	5	0	0	0	0	0	0	0
Moranda (Geometric De-eut)	5	0	0	0	0	0	0	0	0
Moranda (Hybrid Geomet Poiss)	5	0	0	0	0	0	0	0	0
Goel/Okumoto	5	0	0	0	0	0	0	0	0
Littlewood/Verrall	5	0	0	0	0	0	0	0	0
Lloyd/Lipow	5	0	0	0	0	0	0	0	0
Complexity									
Halstead	5	0	0	0	0	5	0	0	0
McCabe	5	0	0	0	0	5	0	0	0
Woodward (Knot Counts)	5	0	0	0	0	5	0	0	0
Chen (Nested Decision Stmts)	5	0	0	0	0	5	0	0	0
Gaffney	5	0	0	0	0	5	0	0	0
Benyon-Tinker	5	0	0	0	0	5	0	0	0
Gilb's (Binary Decision)	5	0	0	0	0	5	0	0	0
Chapin's Q	5	0	0	0	0	5	0	0	0
Segment-Global Usage Pair	5	0	0	0	0	5	0	0	0
Myer's (McCabe extension)	5	0	0	0	0	5	0	0	0
Hansen's (McCabe/Halstead)	5	0	0	0	0	5	0	0	0
Oviedo's (Data/Ctrl Flows)	5	0	0	0	0	5	0	0	0
Failure Count									
Geol/Okumoto	5	0	0	0	0	0	0	0	0
Schneidewind	5	0	0	0	0	0	0	0	0
Geol	5	0	0	0	0	0	0	0	0
Musa	5	0	0	0	0	0	0	0	0
Shooman	5	0	0	0	0	0	0	0	0
Moranda	5	5	0	0	0	0	0	0	0
Jelinski/Moranda	5	5	0	0	0	0	0	0	0
Moranda	5	5	0	0	0	0	0	0	0
Schick/Wolverton	5	5	0	0	0	0	0	0	0
Goel/Okumoto (Gen Poisson)	5	5	0	0	0	0	0	0	0
Brooks/Motley	5	5	0	0	0	0	0	0	0
IBM Poisson	5	5	0	0	0	0	0	0	0
Fault Seeding									
Mills	5	0	0	0	0	0	0	0	0
Input Domain Based									
Nelson	5	0	0	0	0	0	0	0	0
Ho	5	0	0	0	0	0	0	0	0
Ramamoorthy/Bastani	5	0	0	0	0	0	0	0	0

Table B-20 Subfunction Ranked Attributes & Metric Models

Subfunction Title: Command Environment Control

METRIC CLASSES & MODELS	SUBFUNCTION ATTRIBUTES, RANKINGS & SCORES								
	Reli	Surv	Intg	Thru	Usab	Main	Port	Reus	Efc
	H	M	H	M	M	H	M	L	M
Time Between Failure	5	3	5	3	3	5	3	1	3
Jelinski/Moranda	5	3	0	0	0	0	0	0	0
Schick/Wolverton (Linear)	5	3	0	0	0	0	0	0	0
Schick/Wolverton (Parabolic)	5	3	0	0	0	0	0	0	0
Moranda (Geometric De-eut)	5	0	0	0	0	0	0	0	0
Moranda (Hybrid Geomet Poiss)	5	0	0	0	0	0	0	0	0
Goel/Okumoto	5	0	0	0	0	0	0	0	0
Littlewood/Verrall	5	0	0	0	0	0	0	0	0
Lloyd/Lipow	5	0	0	0	0	0	0	0	0
Complexity									
Halstead	5	0	0	0	0	5	0	0	0
McCabe	5	0	0	0	0	5	0	0	0
Woodward (Knot Counts)	5	0	0	0	0	5	0	0	0
Chen (Nested Decision Stmts)	5	0	0	0	0	5	0	0	0
Gaffney	5	0	0	0	0	5	0	0	0
Benyon-Tinker	5	0	0	0	0	5	0	0	0
Gilb's (Binary Decision)	5	0	0	0	0	5	0	0	0
Chapin's Q	5	0	0	0	0	5	0	0	0
Segment-Global Usage Pair	5	0	0	0	0	5	0	0	0
Myer's (McCabe extension)	5	0	0	0	0	5	0	0	0
Hansen's (McCabe/Halstead)	5	0	0	0	0	5	0	0	0
Oviedo's (Data/Ctrl Flows)	5	0	0	0	0	5	0	0	0
Failure Count									
Geol/Okumoto	5	0	0	0	0	0	0	0	0
Schneidewind	5	0	0	0	0	0	0	0	0
Geol	5	0	0	0	0	0	0	0	0
Musa	5	0	0	0	0	0	0	0	0
Shooman	5	0	0	0	0	0	0	0	0
Moranda	5	3	0	0	0	0	0	0	0
Jelinski/Moranda	5	3	0	0	0	0	0	0	0
Moranda	5	3	0	0	0	0	0	0	0
Schick/Wolverton	5	3	0	0	0	0	0	0	0
Goel/Okumoto (Gen Poisson)	5	3	0	0	0	0	0	0	0
Brooks/Motley	5	3	0	0	0	0	0	0	0
IBM Poisson	5	3	0	0	0	0	0	0	0
Fault Seeding									
Mills	5	0	0	0	0	0	0	0	0
Input Domain Based									
Nelson	5	0	0	0	0	0	0	0	0
Ho	5	0	0	0	0	0	0	0	0
Ramamoorthy/Bastani	5	0	0	0	0	0	0	0	0

Table B-21 Subfunction Ranked Attributes & Metric Models

Subfunction Title: Control Onboard Environment

METRIC CLASSES & MODELS	SUBFUNCTION ATTRIBUTES, RANKINGS & SCORES								
	Reli	Surv	Intg	Thru	Usab	Main	Port	Reus	Effc
	H	H	H	H	L	L	L	L	M
Time Between Failure	5	5	5	5	1	1	1	1	3
Jelinski/Moranda	5	5	0	0	0	0	0	0	0
Schick/Wolverton (Linear)	5	5	0	0	0	0	0	0	0
Schick/Wolverton (Parabolic)	5	5	0	0	0	0	0	0	0
Moranda (Geometric De-eut)	5	0	0	0	0	0	0	0	0
Moranda (Hybrid Geomet Poiss)	5	0	0	0	0	0	0	0	0
Goel/Okumoto	5	0	0	0	0	0	0	0	0
Littlewood/Verrall	5	0	0	0	0	0	0	0	0
Lloyd/Lipow	5	0	0	0	0	0	0	0	0
Complexity									
Halstead	5	0	0	0	0	1	0	0	0
McCabe	5	0	0	0	0	1	0	0	0
Woodward (Knot Counts)	5	0	0	0	0	1	0	0	0
Chen (Nested Decision Stmts)	5	0	0	0	0	1	0	0	0
Gaffney	5	0	0	0	0	1	0	0	0
Benyon-Tinker	5	0	0	0	0	1	0	0	0
Gilb's (Binary Decision)	5	0	0	0	0	1	0	0	0
Chapin's Q	5	0	0	0	0	1	0	0	0
Segment-Global Usage Pair	5	0	0	0	0	1	0	0	0
Myer's (McCabe extension)	5	0	0	0	0	1	0	0	0
Hansen's (McCabe/Halstead)	5	0	0	0	0	1	0	0	0
Oviedo's (Data/Ctrl Flows)	5	0	0	0	0	1	0	0	0
Failure Count									
Geol/Okumoto	5	0	0	0	0	0	0	0	0
Schneidewind	5	0	0	0	0	0	0	0	C
Geol	5	0	0	0	0	0	0	0	0
Musa	5	0	0	0	0	0	0	0	0
Shooman	5	0	0	0	0	0	0	0	0
Moranda	5	5	0	0	0	0	0	0	0
Jelinski/Moranda	5	5	0	0	0	0	0	0	0
Moranda	5	5	0	0	0	0	0	0	0
Schick/Wolverton	5	5	0	0	0	0	0	0	0
Goel/Okumoto (Gen Poisson)	5	5	0	0	0	0	0	0	0
Brooks/Motley	5	5	0	0	0	0	0	0	0
IBM Poisson	5	5	0	0	0	0	0	0	0
Fault Seeding									
Mills	5	0	0	0	0	0	0	0	0
Input Domain Based									
Nelson	5	0	0	0	0	0	0	0	0
Ho	5	0	0	0	0	0	0	0	0
Ramamoorthy/Bastani	5	0	0	0	0	0	0	0	0

Table B-22 Subfunction Ranked Attributes & Metric Models

Subfunction Title: Command Attitude and Position Control

METRIC CLASSES & MODELS	SUBFUNCTION ATTRIBUTES, RANKINGS & SCORES								
	Reli	Surv	Intg	Thru	Usab	Main	Port	Reus	Effc
	H	M	H	M	M	H	M	L	M
Time Between Failure	5	3	5	3	3	5	3	1	3
Jelinski/Moranda	5	3	0	0	0	0	0	0	0
Schick/Wolverton (Linear)	5	3	0	0	0	0	0	0	0
Schick/Wolverton (Parabolic)	5	3	0	0	0	0	0	0	0
Moranda (Geometric De-eut)	5	0	0	0	0	0	0	0	0
Moranda (Hybrid Geomet Poiss)	5	0	0	0	0	0	0	0	0
Goel/Okumoto	5	0	0	0	0	0	0	0	0
Littlewood/Verrall	5	0	0	0	0	0	0	0	0
Lloyd/Lipow	5	0	0	0	0	0	0	0	0
Complexity									
Halstead	5	0	0	0	0	5	0	0	0
McCabe	5	0	0	0	0	5	0	0	0
Woodward (Knot Counts)	5	0	0	0	0	5	0	0	0
Chen (Nested Decision Stmts)	5	0	0	0	0	5	0	0	0
Gaffney	5	0	0	0	0	5	0	0	0
Benyon-Tinker	5	0	0	0	0	5	0	0	0
Gilb's (Binary Decision)	5	0	0	0	0	5	0	0	0
Chapin's Q	5	0	0	0	0	5	0	0	0
Segment-Global Usage Pair	5	0	0	0	0	5	0	0	0
Myer's (McCabe extension)	5	0	0	0	0	5	0	0	0
Hansen's (McCabe/Halstead)	5	0	0	0	0	5	0	0	0
Oviedo's (Data/Ctrl Flows)	5	0	0	0	0	5	0	0	0
Failure Count									
Geol/Okumoto	5	0	0	0	0	0	0	0	0
Schneidewind	5	0	0	0	0	0	0	0	0
Geol	5	0	0	0	0	0	0	0	0
Musa	5	0	0	0	0	0	0	0	0
Shooman	5	0	0	0	0	0	0	0	0
Moranda	5	3	0	0	0	0	0	0	0
Jelinski/Moranda	5	3	0	0	0	0	0	0	0
Moranda	5	3	0	0	0	0	0	0	0
Schick/Wolverton	5	3	0	0	0	0	0	0	0
Goel/Okumoto (Gen Poisson)	5	3	0	0	0	0	0	0	0
Brooks/Motley	5	3	0	0	0	0	0	0	0
IBM Poisson	5	3	0	0	0	0	0	0	0
Fault Seeding									
Mills	5	0	0	0	0	0	0	0	0
Input Domain Based									
Nelson	5	0	0	0	0	0	0	0	0
Ho	5	0	0	0	0	0	0	0	0
Ramamoorthy/Bastani	5	0	0	0	0	0	0	0	0

Table B-23 Subfunction Ranked Attributes & Metric Models

Subfunction Title: Control Onboard Attitude and Position

METRIC CLASSES & MODELS	SUBFUNCTION ATTRIBUTES, RANKINGS & SCORES								
	Reli	Surv	Intg	Thru	Usab	Main	Port	Reus	Effc
	H	H	H	H	L	L	L	L	M
Time Between Failure	5	5	5	5	1	1	1	1	3
Jelinski/Moranda	5	5	0	0	0	0	0	0	0
Schick/Wolverton (Linear)	5	5	0	0	0	0	0	0	0
Schick/Wolverton (Parabolic)	5	5	0	0	0	0	0	0	0
Moranda (Geometric De-eut)	5	0	0	0	0	0	0	0	0
Moranda (Hybrid Geomet Poiss)	5	0	0	0	0	0	0	0	0
Goel/Okumoto	5	0	0	0	0	0	0	0	0
Littlewood/Verrall	5	0	0	0	0	0	0	0	0
Lloyd/Lipow	5	0	0	0	0	0	0	0	0
Complexity									
Halstead	5	0	0	0	0	1	0	0	0
McCabe	5	0	0	0	0	1	0	0	0
Woodward (Knot Counts)	5	0	0	0	0	1	0	0	0
Chen (Nested Decision Stmts)	5	0	0	0	0	1	0	0	0
Gaffney	5	0	0	0	0	1	0	0	0
Benyon-Tinker	5	0	0	0	0	1	0	0	0
Gilb's (Binary Decision)	5	0	0	0	0	1	0	0	0
Chapin's Q	5	0	0	0	0	1	0	0	0
Segment-Global Usage Pair	5	0	0	0	0	1	0	0	0
Myer's (McCabe extension)	5	0	0	0	0	1	0	0	0
Hansen's (McCabe/Halstead)	5	0	0	0	0	1	0	0	0
Oviedo's (Data/Ctrl Flows)	5	0	0	0	0	1	0	0	0
Failure Count									
Geol/Okumoto	5	0	0	0	0	0	0	0	0
Schneidewind	5	0	0	0	0	0	0	0	0
Geol	5	0	0	0	0	0	0	0	0
Musa	5	0	0	0	0	0	0	0	0
Shooman	5	0	0	0	0	0	0	0	0
Moranda	5	5	0	0	0	0	0	0	0
Jelinski/Moranda	5	5	0	0	0	0	0	0	0
Moranda	5	5	0	0	0	0	0	0	0
Schick/Wolverton	5	5	0	0	0	0	0	0	0
Goel/Okumoto (Gen Poisson)	5	5	0	0	0	0	0	0	0
Brooks/Motley	5	5	0	0	0	0	0	0	0
IBM Poisson	5	5	0	0	0	0	0	0	0
Fault Seeding									
Mills	5	0	0	0	0	0	0	0	0
Input Domain Based									
Nelson	5	0	0	0	0	0	0	0	0
Ho	5	0	0	0	0	0	0	0	0
Ramamoorthy/Bastani	5	0	0	0	0	0	0	0	0

Table B-24 Subfunction Ranked Attributes & Metric Models

Subfunction Title: Sense Onboard Status

METRIC CLASSES & MODELS	SUBFUNCTION ATTRIBUTES, RANKINGS & SCORES								
	Reli	Surv	Intg	Thru	Usab	Main	Port	Reus	Effc
	H	H	H	M	L	L	L	L	M
Time Between Failure	5	5	5	3	1	1	1	1	3
Jelinski/Moranda	5	5	0	0	0	0	0	0	0
Schick/Wolverton (Linear)	5	5	0	0	0	0	0	0	0
Schick/Wolverton (Parabolic)	5	5	0	0	0	0	0	0	0
Moranda (Geometric De-eut)	5	0	0	0	0	0	0	0	0
Moranda (Hybrid Geomet Poiss)	5	0	0	0	0	0	0	0	0
Goel/Okumoto	5	0	0	0	0	0	0	0	0
Littlewood/Verrall	5	0	0	0	0	0	0	0	0
Lloyd/Lipow	5	0	0	0	0	0	0	0	0
Complexity									
Halstead	5	0	0	0	0	1	0	0	0
McCabe	5	0	0	0	0	1	0	0	0
Woodward (Knot Counts)	5	0	0	0	0	1	0	0	0
Chen (Nested Decision Stmts)	5	0	0	0	0	1	0	0	0
Gaffney	5	0	0	0	0	1	0	0	0
Benyon-Tinker	5	0	0	0	0	1	0	0	0
Gilb's (Binary Decision)	5	0	0	0	0	1	0	0	0
Chapin's Q	5	0	0	0	0	1	0	0	0
Segment-Global Usage Pair	5	0	0	0	0	1	0	0	0
Myer's (McCabe extension)	5	0	0	0	0	1	0	0	0
Hansen's (McCabe/Halstead)	5	0	0	0	0	1	0	0	0
Oviedo's (Data/Ctrl Flows)	5	0	0	0	0	1	0	0	0
Failure Count									
Geol/Okumoto	5	0	0	0	0	0	0	0	0
Schneidewind	5	0	0	0	0	0	0	0	0
Geol	5	0	0	0	0	0	0	0	0
Musa	5	0	0	0	0	0	0	0	0
Shooman	5	0	0	0	0	0	0	0	0
Moranda	5	5	0	0	0	0	0	0	0
Jelinski/Moranda	5	5	0	0	0	0	0	0	0
Moranda	5	5	0	0	0	0	0	0	0
Schick/Wolverton	5	5	0	0	0	0	0	0	0
Goel/Okumoto (Gen Poisson)	5	5	0	0	0	0	0	0	0
Brooks/Motley	5	5	0	0	0	0	0	0	0
IBM Poisson	5	5	0	0	0	0	0	0	0
Fault Seeding									
Mills	5	0	0	0	0	0	0	0	0
Input Domain Based									
Nelson	5	0	0	0	0	0	0	0	0
Ho	5	0	0	0	0	0	0	0	0
Ramamoorthy/Bastani	5	0	0	0	0	0	0	0	0

Table B-25 Subfunction Ranked Attributes & Metric Models

Subfunction Title: Assess Status

METRIC CLASSES & MODELS	SUBFUNCTION ATTRIBUTES, RANKINGS & SCORES								
	Reli	Surv	Intg	Thru	Usab	Main	Port	Reus	Effc
	H	M	H	M	M	H	M	L	M
Time Between Failure	5	3	5	3	3	5	3	1	3
Jelinski/Moranda	5	3	0	0	0	0	0	0	0
Schick/Wolverton (Linear)	5	3	0	0	0	0	0	0	0
Schick/Wolverton (Parabolic)	5	3	0	0	0	0	0	0	0
Moranda (Geometric De-eut)	5	0	0	0	0	0	0	0	0
Moranda (Hybrid Geomet Poiss)	5	0	0	0	0	0	0	0	0
Goel/Okumoto	5	0	0	0	0	0	0	0	0
Littlewood/Verrall	5	0	0	0	0	0	0	0	0
Lloyd/Lipow	5	0	0	0	0	0	0	0	0
Complexity									
Halstead	5	0	0	0	0	5	0	0	0
McCabe	5	0	0	0	0	5	0	0	0
Woodward (Knot Counts)	5	0	0	0	0	5	0	0	0
Chen (Nested Decision Stmts)	5	0	0	0	0	5	0	0	0
Gaffney	5	0	0	0	0	5	0	0	0
Benyon-Tinker	5	0	0	0	0	5	0	0	0
Gilb's (Binary Decision)	5	0	0	0	0	5	0	0	0
Chapin's Q	5	0	0	0	0	5	0	0	0
Segment-Global Usage Pair	5	0	0	0	0	5	0	0	0
Myer's (McCabe extension)	5	0	0	0	0	5	0	0	0
Hansen's (McCabe/Halstead)	5	0	0	0	0	5	0	0	0
Oviedo's (Data/Ctrl Flows)	5	0	0	0	0	5	0	0	0
Failure Count									
Geol/Okumoto	5	0	0	0	0	0	0	0	0
Schneidewind	5	0	0	0	0	0	0	0	0
Geol	5	0	0	0	0	0	0	0	0
Musa	5	0	0	0	0	0	0	0	0
Shooman	5	0	0	0	0	0	0	0	0
Moranda	5	3	0	0	0	0	0	0	0
Jelinski/Moranda	5	3	0	0	0	0	0	0	0
Moranda	5	3	0	0	0	0	0	0	0
Schick/Wolverton	5	3	0	0	0	0	0	0	0
Goel/Okumoto (Gen Poisson)	5	3	0	0	0	0	0	0	0
Brooks/Motley	5	3	0	0	0	0	0	0	0
IBM Poisson	5	3	0	0	0	0	0	0	0
Fault Seeding									
Mills	5	0	0	0	0	0	0	0	0
Input Domain Based									
Nelson	5	0	0	0	0	0	0	0	0
Ho	5	0	0	0	0	0	0	0	0
Ramamoorthy/Bastani	5	0	0	0	0	0	0	0	0

Table B-26 Subfunction Ranked Attributes & Metric Models

Subfunction Title: Command Reconfiguration

METRIC CLASSES & MODELS	SUBFUNCTION ATTRIBUTES, RANKINGS & SCORES								
	Reli	Surv	Intg	Thru	Usab	Main	Port	Reus	Effc
	H	M	H	M	M	H	M	L	M
Time Between Failure	5	3	5	3	3	5	3	1	3
Jelinski/Moranda	5	3	0	0	0	0	0	0	0
Schick/Wolverton (Linear)	5	3	0	0	0	0	0	0	0
Schick/Wolverton (Parabolic)	5	3	0	0	0	0	0	0	0
Moranda (Geometric De-eut)	5	0	0	0	0	0	0	0	0
Moranda (Hybrid Geomet Poiss)	5	0	0	0	0	0	0	0	0
Goel/Okumoto	5	0	0	0	0	0	0	0	0
Littlewood/Verrall	5	0	0	0	0	0	0	0	0
Lloyd/Lipow	5	0	0	0	0	0	0	0	0
Complexity									
Halstead	5	0	0	0	0	5	0	0	0
McCabe	5	0	0	0	0	5	0	0	0
Woodward (Knot Counts)	5	0	0	0	0	5	0	0	0
Chen (Nested Decision Stmts)	5	0	0	0	0	5	0	0	0
Gaffney	5	0	0	0	0	5	0	0	0
Benyon-Tinker	5	0	0	0	0	5	0	0	0
Gilb's (Binary Decision)	5	0	0	0	0	5	0	0	0
Chapin's Q	5	0	0	0	0	5	0	0	0
Segment-Global Usage Pair	5	0	0	0	0	5	0	0	0
Myer's (McCabe extension)	5	0	0	0	0	5	0	0	0
Hansen's (McCabe/Halstead)	5	0	0	0	0	5	0	0	0
Oviedo's (Data/Ctrl Flows)	5	0	0	0	0	5	0	0	0
Failure Count									
Geol/Okumoto	5	0	0	0	0	0	0	0	0
Schneidewind	5	0	0	0	0	0	0	0	0
Geol	5	0	0	0	0	0	0	0	0
Musa	5	0	0	0	0	0	0	0	0
Shooman	5	0	0	0	0	0	0	0	0
Moranda	5	3	0	0	0	0	0	0	0
Jelinski/Moranda	5	3	0	0	0	0	0	0	0
Moranda	5	3	0	0	0	0	0	0	0
Schick/Wolverton	5	3	0	0	0	0	0	0	0
Goel/Okumoto (Gen Poisson)	5	3	0	0	0	0	0	0	0
Brooks/Motley	5	3	0	0	0	0	0	0	0
IBM Poisson	5	3	0	0	0	0	0	0	0
Fault Seeding									
Mills	5	0	0	0	0	0	0	0	0
Input Domain Based									
Nelson	5	0	0	0	0	0	0	0	0
Ho	5	0	0	0	0	0	0	0	0
Ramamoorthy/Bastani	5	0	0	0	0	0	0	0	0

Table B-27 Subfunction Ranked Attributes & Metric Models

Subfunction Title: Reconfiguration

METRIC CLASSES & MODELS	SUBFUNCTION ATTRIBUTES, RANKINGS & SCORES								
	Reli	Surv	Intg	Thru	Usab	Main	Port	Reus	Effc
	H	H	H	H	L	L	L	L	M
Time Between Failure	5	5	5	5	1	1	1	1	3
Jelinski/Moranda	5	5	0	0	0	0	0	0	0
Schick/Wolverton (Linear)	5	5	0	0	0	0	0	0	0
Schick/Wolverton (Parabolic)	5	5	0	0	0	0	0	0	0
Moranda (Geometric De-eut)	5	0	0	0	0	0	0	0	0
Moranda (Hybrid Geomet Poiss)	5	0	0	0	0	0	0	0	0
Goel/Okumoto	5	0	0	0	0	0	0	0	0
Littlewood/Verrall	5	0	0	0	0	0	0	0	0
Lloyd/Lipow	5	0	0	0	0	0	0	0	0
Complexity									
Halstead	5	0	0	0	0	1	0	0	0
McCabe	5	0	0	0	0	1	0	0	0
Woodward (Knot Counts)	5	0	0	0	0	1	0	0	0
Chen (Nested Decision Stmts)	5	0	0	0	0	1	0	0	0
Gaffney	5	0	0	0	0	1	0	0	0
Benyon-Tinker	5	0	0	0	0	1	0	0	0
Gilb's (Binary Decision)	5	0	0	0	0	1	0	0	0
Chapin's Q	5	0	0	0	0	1	0	0	0
Segment-Global Usage Pair	5	0	0	0	0	1	0	0	0
Myer's (McCabe extension)	5	0	0	0	0	1	0	0	0
Hansen's (McCabe/Halstead)	5	0	0	0	0	1	0	0	0
Oviedo's (Data/Ctrl Flows)	5	0	0	0	0	1	0	0	0
Failure Count									
Geol/Okumoto	5	0	0	0	0	0	0	0	0
Schneidewind	5	0	0	0	0	0	0	0	0
Geol	5	0	0	0	0	0	0	0	0
Musa	5	0	0	0	0	0	0	0	0
Shooman	5	0	0	0	0	0	0	0	0
Moranda	5	5	0	0	0	0	0	0	0
Jelinski/Moranda	5	5	0	0	0	0	0	0	0
Moranda	5	5	0	0	0	0	0	0	0
Schick/Wolverton	5	5	0	0	0	0	0	0	0
Goel/Okumoto (Gen Poisson)	5	5	0	0	0	0	0	0	0
Brooks/Motley	5	5	0	0	0	0	0	0	0
IBM Poisson	5	5	0	0	0	0	0	0	0
Fault Seeding									
Mills	5	0	0	0	0	0	0	0	0
Input Domain Based									
Nelson	5	0	0	0	0	0	0	0	0
Ho	5	0	0	0	0	0	0	0	0
Ramamoorthy/Bastani	5	0	0	0	0	0	0	0	0

Table B-28 Subfunction Ranked Attributes & Metric Models

Subfunction Title: Development Tools

METRIC CLASSES & MODELS	SUBFUNCTION ATTRIBUTES, RANKINGS & SCORES								
	Reli	Surv	Intg	Thru	Usab	Main	Port	Reus	Effc
	M	M	M	L	M	M	M	M	M
Time Between Failure	3	3	3	1	3	3	3	3	3
Jelinski/Moranda	3	3	0	0	0	0	0	0	0
Schick/Wolverton (Linear)	3	3	0	0	0	0	0	0	0
Schick/Wolverton (Parabolic)	3	3	0	0	0	0	0	0	0
Moranda (Geometric De-eut)	3	0	0	0	0	0	0	0	0
Moranda (Hybrid Geomet Poiss)	3	0	0	0	0	0	0	0	0
Goel/Okumoto	3	0	0	0	0	0	0	0	0
Littlewood/Verrall	3	0	0	0	0	0	0	0	0
Lloyd/Lipow	3	0	0	0	0	0	0	0	0
Complexity									
Halstead	3	0	0	0	0	3	0	0	0
McCabe	3	0	0	0	0	3	0	0	0
Woodward (Knot Counts)	3	0	0	0	0	3	0	0	0
Chen (Nested Decision Stmts)	3	0	0	0	0	3	0	0	0
Gaffney	3	0	0	0	0	3	0	0	0
Benyon-Tinker	3	0	0	0	0	3	0	0	0
Gilb's (Binary Decision)	3	0	0	0	0	3	0	0	0
Chapin's Q	3	0	0	0	0	3	0	0	0
Segment-Global Usage Pair	3	0	0	0	0	3	0	0	0
Myer's (McCabe extension)	3	0	0	0	0	3	0	0	0
Hansen's (McCabe/Halstead)	3	0	0	0	0	3	0	0	0
Oviedo's (Data/Ctrl Flows)	3	0	0	0	0	3	0	0	0
Failure Count									
Geol/Okumoto	3	0	0	0	0	0	0	0	0
Schneidewind	3	0	0	0	0	0	0	0	0
Geol	3	0	0	0	0	0	0	0	0
Musa	3	0	0	0	0	0	0	0	0
Shooman	3	0	0	0	0	0	0	0	0
Moranda	3	3	0	0	0	0	0	0	0
Jelinski/Moranda	3	3	0	0	0	0	0	0	0
Moranda	3	3	0	0	0	0	0	0	0
Schick/Wolverton	3	3	0	0	0	0	0	0	0
Goel/Okumoto (Gen Poisson)	3	3	0	0	0	0	0	0	0
Brooks/Motley	3	3	0	0	0	0	0	0	0
IBM Poisson	3	3	0	0	0	0	0	0	0
Fault Seeding									
Mills	3	0	0	0	0	0	0	0	0
Input Domain Based									
Nelson	3	0	0	0	0	0	0	0	0
Ho	3	0	0	0	0	0	0	0	0
Ramamoorthy/Bastani	3	0	0	0	0	0	0	0	0

Table B-29 Subfunction Ranked Attributes & Metric Models

Subfunction Title: Hardware-in-the-loop (HWIL) Simulation

METRIC CLASSES & MODELS	SUBFUNCTION ATTRIBUTES, RANKINGS & SCORES								
	Reli	Surv	Intg	Thru	Usab	Main	Port	Reus	Effc
	M	H	H	M	M	M	L	L	M
Time Between Failure	3	5	5	3	3	3	1	1	3
Jelinski/Moranda	3	5	0	0	0	0	0	0	0
Schick/Wolverton (Linear)	3	5	0	0	0	0	0	0	0
Schick/Wolverton (Parabolic)	3	5	0	0	0	0	0	0	0
Moranda (Geometric De-eut)	3	0	0	0	0	0	0	0	0
Moranda (Hybrid Geomet Poiss)	3	0	0	0	0	0	0	0	0
Goel/Okumoto	3	0	0	0	0	0	0	0	0
Littlewood/Verrall	3	0	0	0	0	0	0	0	C
Lloyd/Lipow	3	0	0	0	0	0	0	0	0
Complexity									
Halstead	3	0	0	0	0	3	0	0	0
McCabe	3	0	0	0	0	3	0	0	0
Woodward (Knot Counts)	3	0	0	0	0	3	0	0	0
Chen (Nested Decision Stmts)	3	0	0	0	0	3	0	0	0
Gaffney	3	0	0	0	0	3	0	0	0
Benyon-Tinker	3	0	0	0	0	3	0	0	0
Gilb's (Binary Decision)	3	0	0	0	0	3	0	0	0
Chapin's Q	3	0	0	0	0	3	0	0	0
Segment-Global Usage Pair	3	0	0	0	0	3	0	0	0
Myer's (McCabe extension)	3	0	0	0	0	3	0	0	0
Hansen's (McCabe/Halstead)	3	0	0	0	0	3	0	0	0
Oviedo's (Data/Ctrl Flows)	3	0	0	0	0	3	0	0	0
Failure Count									
Geol/Okumoto	3	0	0	0	0	0	0	0	0
Schneidewind	3	0	0	0	0	0	0	0	0
Geol	3	0	0	0	0	0	0	0	0
Musa	3	0	0	0	0	0	0	0	0
Shooman	3	0	0	0	0	0	0	0	0
Moranda	3	5	0	0	0	0	0	0	0
Jelinski/Moranda	3	5	0	0	0	0	0	0	0
Moranda	3	5	0	0	0	0	0	0	0
Schick/Wolverton	3	5	0	0	0	0	0	0	0
Goel/Okumoto (Gen Poisson)	3	5	0	0	0	0	0	0	0
Brooks/Motley	3	5	0	0	0	0	0	0	0
IBM Poisson	3	5	0	0	0	0	0	0	0
Fault Seeding									
Mills	3	0	0	0	0	0	0	0	0
Input Domain Based									
Nelson	3	0	0	0	0	0	0	0	0
Ho	3	0	0	0	0	0	0	0	0
Ramamoorthy/Bastani	3	0	0	0	0	0	0	0	0

Table B-30 Subfunction Ranked Attributes & Metric Models

Subfunction Title: Demonstration Simulation

METRIC CLASSES & MODELS	SUBFUNCTION ATTRIBUTES, RANKINGS & SCORES								
	Reli	Surv	Intg	Thru	Usab	Main	Port	Reus	Effc
	M	H	M	M	M	M	M	L	M
Time Between Failure	3	5	3	3	3	3	3	1	3
Jelinski/Moranda	3	5	0	0	0	0	0	0	0
Schick/Wolverton (Linear)	3	5	0	0	0	0	0	0	0
Schick/Wolverton (Parabolic)	3	5	0	0	0	0	0	0	0
Moranda (Geometric De-eut)	3	0	0	0	0	0	0	0	0
Moranda (Hybrid Geomet Poiss)	3	0	0	0	0	0	0	0	0
Goel/Okumoto	3	0	0	0	0	0	0	0	0
Littlewood/Verrall	3	0	0	0	0	0	0	0	0
Lloyd/Lipow	3	0	0	0	0	0	0	0	0
Complexity									
Halstead	3	0	0	0	0	3	0	0	0
McCabe	3	0	0	0	0	3	0	0	0
Woodward (Knot Counts)	3	0	0	0	0	3	0	0	0
Chen (Nested Decision Stmts)	3	0	0	0	0	3	0	0	0
Gaffney	3	0	0	0	0	3	0	0	0
Benyon-Tinker	3	0	0	0	0	3	0	0	0
Gilb's (Binary Decision)	3	0	0	0	0	3	0	0	0
Chapin's Q	3	0	0	0	0	3	0	0	0
Segment-Global Usage Pair	3	0	0	0	0	3	0	0	0
Myer's (McCabe extension)	3	0	0	0	0	3	0	0	0
Hansen's (McCabe/Halstead)	3	0	0	0	0	3	0	0	0
Oviedo's (Data/Ctrl Flows)	3	0	0	0	0	3	0	0	0
Failure Count									
Geol/Okumoto	3	0	0	0	0	0	0	0	0
Schneidewind	3	0	0	0	0	0	0	0	0
Geol	3	0	0	0	0	0	0	0	0
Musa	3	0	0	0	0	0	0	0	0
Shooman	3	0	0	0	0	0	0	0	0
Moranda	3	5	0	0	0	0	0	0	0
Jelinski/Moranda	3	5	0	0	0	0	0	0	0
Moranda	3	5	0	0	0	0	0	0	0
Schick/Wolverton	3	5	0	0	0	0	0	0	0
Goel/Okumoto (Gen Poisson)	3	5	0	0	0	0	0	0	0
Brooks/Motley	3	5	0	0	0	0	0	0	0
IBM Poisson	3	5	0	0	0	0	0	0	0
Fault Seeding									
Mills	3	0	0	0	0	0	0	0	0
Input Domain Based									
Nelson	3	0	0	0	0	0	0	0	0
Ho	3	0	0	0	0	0	0	0	0
Ramamoorthy/Bastani	3	0	0	0	0	0	0	0	0

Table B-32 Subfunction Ranked Attributes & Metric Models

Subfunction Title: Provide Development Environment

METRIC CLASSES & MODELS	SUBFUNCTION ATTRIBUTES, RANKINGS & SCORES								
	Reli	Surv	Intg	Thru	Usab	Main	Port	Reus	Effc
	H	M	H	M	H	M	L	L	M
Time Between Failure	5	3	5	3	5	3	1	1	3
Jelinski/Moranda	5	3	0	0	0	0	0	0	0
Schick/Wolverton (Linear)	5	3	0	0	0	0	0	0	0
Schick/Wolverton (Parabolic)	5	3	0	0	0	0	0	0	0
Moranda (Geometric De-eut)	5	0	0	0	0	0	0	0	0
Moranda (Hybrid Geomet Poiss)	5	0	0	0	0	0	0	0	0
Goel/Okumoto	5	0	0	0	0	0	0	0	0
Littlewood/Verrall	5	0	0	0	0	0	0	0	0
Lloyd/Lipow	5	0	0	0	0	0	0	0	0
Complexity									
Halstead	5	0	0	0	0	3	0	0	0
McCabe	5	0	0	0	0	3	0	0	0
Woodward (Knot Counts)	5	0	0	0	0	3	0	0	0
Chen (Nested Decision Stmts)	5	0	0	0	0	3	0	0	0
Gaffney	5	0	0	0	0	3	0	0	0
Benyon-Tinker	5	0	0	0	0	3	0	0	0
Gilb's (Binary Decision)	5	0	0	0	0	3	0	0	0
Chapin's Q	5	0	0	0	0	3	0	0	0
Segment-Global Usage Pair	5	0	0	0	0	3	0	0	0
Myer's (McCabe extension)	5	0	0	0	0	3	0	0	0
Hansen's (McCabe/Halstead)	5	0	0	0	0	3	0	0	0
Oviedo's (Data/Ctrl Flows)	5	0	0	0	0	3	0	0	0
Failure Count									
Geol/Okumoto	5	0	0	0	0	0	0	0	0
Schneidewind	5	0	0	0	0	0	0	0	0
Geol	5	0	0	0	0	0	0	0	0
Musa	5	0	0	0	0	0	0	0	0
Shooman	5	0	0	0	0	0	0	0	0
Moranda	5	3	0	0	0	0	0	0	0
Jelinski/Moranda	5	3	0	0	0	0	0	0	0
Moranda	5	3	0	0	0	0	0	0	0
Schick/Wolverton	5	3	0	0	0	0	0	0	0
Goel/Okumoto (Gen Poisson)	5	3	0	0	0	0	0	0	0
Brooks/Motley	5	3	0	0	0	0	0	0	0
IBM Poisson	5	3	0	0	0	0	0	0	0
Fault Seeding									
Mills	5	0	0	0	0	0	0	0	0
Input Domain Based									
Nelson	5	0	0	0	0	0	0	0	0
Ho	5	0	0	0	0	0	0	0	0
Ramamoorthy/Bastani	5	0	0	0	0	0	0	0	0

Table B-33 Subfunction Ranked Attributes & Metric Models

Subfunction Title: Support Factory Test

METRIC CLASSES & MODELS	SUBFUNCTION ATTRIBUTES, RANKINGS & SCORES								
	Reli	Surv	Intg	Thru	Usab	Main	Port	Reus	Effc
	M	M	M	L	H	H	H	H	L
Time Between Failure	3	3	3	1	5	5	5	5	1
Jelinski/Moranda	3	3	0	0	0	0	0	0	0
Schick/Wolverton (Linear)	3	3	0	0	0	0	0	0	0
Schick/Wolverton (Parabolic)	3	3	0	0	0	0	0	0	0
Moranda (Geometric De-eut)	3	0	0	0	0	0	0	0	0
Moranda (Hybrid Geomet Poiss)	3	0	0	0	0	0	0	0	0
Goel/Okumoto	3	0	0	0	0	0	0	0	0
Littlewood/Verrall	3	0	0	0	0	0	0	0	0
Lloyd/Lipow	3	0	0	0	0	0	0	0	0
Complexity									
Halstead	3	0	0	0	0	5	0	0	0
McCabe	3	0	0	0	0	5	0	0	0
Woodward (Knot Counts)	3	0	0	0	0	5	0	0	0
Chen (Nested Decision Stmts)	3	0	0	0	0	5	0	0	0
Gaffney	3	0	0	0	0	5	0	0	0
Benyon-Tinker	3	0	0	0	0	5	0	0	0
Gilb's (Binary Decision)	3	0	0	0	0	5	0	0	0
Chapin's Q	3	0	0	0	0	5	0	0	0
Segment-Global Usage Pair	3	0	0	0	0	5	0	0	0
Myer's (McCabe extension)	3	0	0	0	0	5	0	0	0
Hansen's (McCabe/Halstead)	3	0	0	0	0	5	0	0	0
Oviedo's (Data/Ctrl Flows)	3	0	0	0	0	5	0	0	0
Failure Count									
Geol/Okumoto	3	0	0	0	0	0	0	0	0
Schneidewind	3	0	0	0	0	0	0	0	0
Geol	3	0	0	0	0	0	0	0	0
Musa	3	0	0	0	0	0	0	0	0
Shooman	3	0	0	0	0	0	0	0	0
Moranda	3	3	0	0	0	0	0	0	0
Jelinski/Moranda	3	3	0	0	0	0	0	0	0
Moranda	3	3	0	0	0	0	0	0	0
Schick/Wolverton	3	3	0	0	0	0	0	0	0
Goel/Okumoto (Gen Poisson)	3	3	0	0	0	0	0	0	0
Brooks/Motley	3	3	0	0	0	0	0	0	0
IBM Poisson	3	3	0	0	0	0	0	0	0
Fault Seeding									
Mills	3	0	0	0	0	0	0	0	0
Input Domain Based									
Nelson	3	0	0	0	0	0	0	0	0
Ho	3	0	0	0	0	0	0	0	0
Ramamoorthy/Bastani	3	0	0	0	0	0	0	0	0

Table B-34 Subfunction Ranked Attributes & Metric Models

Subfunction Title: Support Acceptance Test

METRIC CLASSES & MODELS	SUBFUNCTION ATTRIBUTES, RANKINGS & SCORES								
	Reli	Surv	Intg	Thru	Usab	Main	Port	Reus	Effc
	M	M	M	M	L	M	L	L	M
Time Between Failure	3	3	3	3	1	3	1	1	3
Jelinski/Moranda	3	3	0	0	0	0	0	0	0
Schick/Wolverton (Linear)	3	3	0	0	0	0	0	0	0
Schick/Wolverton (Parabolic)	3	3	0	0	0	0	0	0	0
Moranda (Geometric De-eut)	3	0	0	0	0	0	0	0	0
Moranda (Hybrid Geomet Poiss)	3	0	0	0	0	0	0	0	0
Goel/Okumoto	3	0	0	0	0	0	0	0	0
Littlewood/Verrall	3	0	0	0	0	0	0	0	0
Lloyd/Lipow	3	0	0	0	0	0	0	0	0
Complexity									
Halstead	3	0	0	0	0	3	0	0	0
McCabe	3	0	0	0	0	3	0	0	0
Woodward (Knot Counts)	3	0	0	0	0	3	0	0	0
Chen (Nested Decision Stmts)	3	0	0	0	0	3	0	0	0
Gaffney	3	0	0	0	0	3	0	0	0
Benyon-Tinker	3	0	0	0	0	3	0	0	0
Gilb's (Binary Decision)	3	0	0	0	0	3	0	0	0
Chapin's Q	3	0	0	0	0	3	0	0	0
Segment-Global Usage Pair	3	0	0	0	0	3	0	0	0
Myer's (McCabe extension)	3	0	0	0	0	3	0	0	0
Hansen's (McCabe/Halstead)	3	0	0	0	0	3	0	0	0
Oviedo's (Data/Ctrl Flows)	3	0	0	0	0	3	0	0	0
Failure Count									
Geol/Okumoto	3	0	0	0	0	0	0	0	0
Schneidewind	3	0	0	0	0	0	0	0	0
Geol	3	0	0	0	0	0	0	0	0
Musa	3	0	0	0	0	0	0	0	0
Shooman	3	0	0	0	0	0	0	0	0
Moranda	3	3	0	0	0	0	0	0	0
Jelinski/Moranda	3	3	0	0	0	0	0	0	0
Moranda	3	3	0	0	0	0	0	0	0
Schick/Wolverton	3	3	0	0	0	0	0	0	0
Goel/Okumoto (Gen Poisson)	3	3	0	0	0	0	0	0	0
Brooks/Motley	3	3	0	0	0	0	0	0	0
IBM Poisson	3	3	0	0	0	0	0	0	0
Fault Seeding									
Mills	3	0	0	0	0	0	0	0	0
Input Domain Based									
Nelson	3	0	0	0	0	0	0	0	0
Ho	3	0	0	0	0	0	0	0	0
Ramamoorthy/Bastani	3	0	0	0	0	0	0	0	0

Table B-35 Subfunction Ranked Attributes & Metric Models

Subfunction Title: Maintain and Control Management Information Database

METRIC CLASSES & MODELS	SUBFUNCTION ATTRIBUTES, RANKINGS & SCORES								
	Reli	Surv	Intg	Thru	Usab	Main	Port	Reus	Effc
	M	M	H	M	H	H	H	H	M
Time Between Failure	3	3	5	3	5	5	5	5	3
Jelinski/Moranda	3	3	0	0	0	0	0	0	0
Schick/Wolverton (Linear)	3	3	0	0	0	0	0	0	0
Schick/Wolverton (Parabolic)	3	3	0	0	0	0	0	0	0
Moranda (Geometric De-eut)	3	0	0	0	0	0	0	0	0
Moranda (Hybrid Geomet Poiss)	3	0	0	0	0	0	0	0	0
Goel/Okumoto	3	0	0	0	0	0	0	0	0
Littlewood/Verrall	3	0	0	0	0	0	0	0	0
Lloyd/Lipow	3	0	0	0	0	0	0	0	0
Complexity									
Halstead	3	0	0	0	0	5	0	0	0
McCabe	3	0	0	0	0	5	0	0	0
Woodward (Knot Counts)	3	0	0	0	0	5	0	0	0
Chen (Nested Decision Stmts)	3	0	0	0	0	5	0	0	0
Gaffney	3	0	0	0	0	5	0	0	0
Benyon-Tinker	3	0	0	0	0	5	0	0	0
Gilb's (Binary Decision)	3	0	0	0	0	5	0	0	0
Chapin's Q	3	0	0	0	0	5	0	0	0
Segment-Global Usage Pair	3	0	0	0	0	5	0	0	0
Myer's (McCabe extension)	3	0	0	0	0	5	0	0	0
Hansen's (McCabe/Halstead)	3	0	0	0	0	5	0	0	0
Oviedo's (Data/Ctrl Flows)	3	0	0	0	0	5	0	0	0
Failure Count									
Geol/Okumoto	3	0	0	0	0	0	0	0	0
Schneidewind	3	0	0	0	0	0	0	0	0
Geol	3	0	0	0	0	0	0	0	0
Musa	3	0	0	0	0	0	0	0	0
Shooman	3	0	0	0	0	0	0	0	0
Moranda	3	3	0	0	0	0	0	0	0
Jelinski/Moranda	3	3	0	0	0	0	0	0	0
Moranda	3	3	0	0	0	0	0	0	0
Schick/Wolverton	3	3	0	0	0	0	0	0	0
Goel/Okumoto (Gen Poisson)	3	3	0	0	0	0	0	0	0
Brooks/Motley	3	3	0	0	0	0	0	0	0
IBM Poisson	3	3	0	0	0	0	0	0	0
Fault Seeding									
Mills	3	0	0	0	0	0	0	0	0
Input Domain Based									
Nelson	3	0	0	0	0	0	0	0	0
Ho	3	0	0	0	0	0	0	0	0
Ramamoorthy/Bastani	3	0	0	0	0	0	0	0	0

Table B-36 Subfunction Ranked Attributes & Metric Models

Subfunction Title: Management Information Tracking

METRIC CLASSES & MODELS	SUBFUNCTION ATTRIBUTES, RANKINGS & SCORES								
	Reli	Surv	Intg	Thru	Usab	Main	Port	Reus	Effc
	M	L	H	L	H	H	H	H	M
Time Between Failure	3	1	5	1	5	5	5	5	3
Jelinski/Moranda	3	1	0	0	0	0	0	0	0
Schick/Wolverton (Linear)	3	1	0	0	0	0	0	0	0
Schick/Wolverton (Parabolic)	3	1	0	0	0	0	0	0	0
Moranda (Geometric De-eut)	3	0	0	0	0	0	0	0	0
Moranda (Hybrid Geomet Poiss)	3	0	0	0	0	0	0	0	0
Goel/Okumoto	3	0	0	0	0	0	0	0	0
Littlewood/Verrall	3	0	0	0	0	0	0	0	0
Lloyd/Lipow	3	0	0	0	0	0	0	0	0
Complexity									
Halstead	3	0	0	0	0	5	0	0	0
McCabe	3	0	0	0	0	5	0	0	0
Woodward (Knot Counts)	3	0	0	0	0	5	0	0	0
Chen (Nested Decision Stmts)	3	0	0	0	0	5	0	0	0
Gaffney	3	0	0	0	0	5	0	0	0
Benyon-Tinker	3	0	0	0	0	5	0	0	0
Gilb's (Binary Decision)	3	0	0	0	0	5	0	0	0
Chapin's Q	3	0	0	0	0	5	0	0	0
Segment-Global Usage Pair	3	0	0	0	0	5	0	0	0
Myer's (McCabe extension)	3	0	0	0	0	5	0	0	0
Hansen's (McCabe/Halstead)	3	0	0	0	0	5	0	0	0
Oviedo's (Data/Ctrl Flows)	3	0	0	0	0	5	0	0	0
Failure Count									
Geol/Okumoto	3	0	0	0	0	0	0	0	0
Schneidewind	3	0	0	0	0	0	0	0	0
Geol	3	0	0	0	0	0	0	0	0
Musa	3	0	0	0	0	0	0	0	0
Shooman	3	0	0	0	0	0	0	0	0
Moranda	3	1	0	0	0	0	0	0	0
Jelinski/Moranda	3	1	0	0	0	0	0	0	0
Moranda	3	1	0	0	0	0	0	0	0
Schick/Wolverton	3	1	0	0	0	0	0	0	0
Goel/Okumoto (Gen Poisson)	3	1	0	0	0	0	0	0	0
Brooks/Motley	3	1	0	0	0	0	0	0	0
IBM Poisson	3	1	0	0	0	0	0	0	0
Fault Seeding									
Mills	3	0	0	0	0	0	0	0	0
Input Domain Based									
Nelson	3	0	0	0	0	0	0	0	0
Ho	3	0	0	0	0	0	0	0	0
Ramamoorthy/Bastani	3	0	0	0	0	0	0	0	0

APPENDIX C

BROAD SELECTION OF SOFTWARE/ ENVIRONMENT SUPPORT TOOLS

C-1 through C-20

LEGEND

<COLUMN HEADING> "Field Value"	DESCRIPTION
<APPL-N>	Major Category
1. "SWAT"	o Software Analysis Tool
2. "PSM"	o Project Setup/Monitor
3. "OLEA"	o On Line Expert Assistance
<TYPE>	Sub Category
	(Meaning Full Functional Descriptors are Used)
<TOOL-NAM>	Tool Name
<COMPANY>	(Self Explanatory)
<CITY>	"
<STATE>	"

Technology for the Assessment of Software Quality "TASQ"

APPL_N	TYPE	TOOL_NAME	COMPANY	CITY & STATE		IBM	VAX	SUN	APOLLO	IBM	MAC
SWAT	APSE	ADA DEV. ENV	DATA GEN. CORP	WESTBORO	MA						
SWAT	ASSEMBLER	MICRO ASSEMB	MICROTEC RESEA	SANTA CLARA	CA		V	S	A	P	
SWAT	ASSEMBLER	MACRO ASSEM	PSS	SANTA MONICA	CA						
SWAT	ASSEMBLER	FRAMEWORK AS	SPECIALIZED SYS	SEATTLE	WA						
SWAT	ASSEMBLER	ASSEMBLER	2500 AD S/W	BUENAVISTA	CA		V	S	A	P	
SWAT	ASSEMBLER	ASSEM-LINK	OASYS	WALTHAM	MA		V	S	A		
SWAT	ASSEMBLER	1750A ASSEM	PSS	SANTA MONICA	CA						
SWAT	AUTO TEST GE	TDGEN	S/W RESEARCH IN	SAN FRANCISCO	CA		V			P	
SWAT	AUTOTESTGEN	T	PROGRAMMING ENV	TINTON FALL	NJ					P	
SWAT	CASE	XL/PROGRAMME	INDEX TECHNOLOG	CAMBRIDGE	MA		V	S	A	P	
SWAT	CASE	VISIBLE ANAL	VISIBLE SYS COR	NEWTON	MA					P	
SWAT	CASE	TIS/XA	CINCOM	NORCROSS	GA	I				P	
SWAT	CASE	TEAMWORK/SD	CADRE	PROVIDENCE	RI		V	S	A		
SWAT	CASE	TEAMWORK/SA	CADRE	PROVIDENCE	RI		V	S	A		
SWAT	CASE	TEAMWORK/RT	CADRE	PROVIDENCE	RI		V	S	A		
SWAT	CASE	TEAMWORK/PCS	CADRE	PROVIDENCE	RI					P	
SWAT	CASE	TEAMWORK/IM	CADRE	PROVIDENCE	RI		V	S	A		
SWAT	CASE	TEAMWORK/ADA	CADRE	PROVIDENCE	RI		V	S	A		
SWAT	CASE	TEAMWORK/ACC	CADRE	PROVIDENCE	RI		V	S	A		
SWAT	CASE	TAGS	TELEDYNE BROWN	HUNTSVILLE	AL		V	S	A		
SWAT	CASE	SUPERCASE	ADV TECH INTER	NEW YORK	NY		V				
SWAT	CASE	SUPER PDL	ADV TECH INTER	NEW YORK	NY		V				
SWAT	CASE	SOURCE VIEW	HEARTLAND	LAWRENCE	KS						
SWAT	CASE	SMALL TALK80	PARCPLACE SYSTE	PALO ALTO	CA	I		S	A		M
SWAT	CASE	SDD	ORACLE CORP	BELMONT	CA	I	V				
SWAT	CASE	SAW	MICROCASE	VALRICO	FL			S		P	
SWAT	CASE	S/W T/PICTUR	INTERACTIVE DEV	NATICK	MA		V	S	A		
SWAT	CASE	PSL/PSA	META SYSTEMS	ANN ARBOR	MI	I	V			P	
SWAT	CASE	PROMOD/SA	PROMOD	LAGUNA HILLS	CA		V			P	
SWAT	CASE	PROMOD/RT	PROMOD	LAGUNA HILLS	CA		V			P	

Technology for the Assessment of Software Quality "TASQ"

APPL_N	TYPE	TOOL_NAME	COMPANY	CITY & STATE		IBM	VAX	SUN	APOLLO	IBM	MAC
SWAT	CASE	PROMOD/MD	PROMOD	LAGUNA HILLS	CA		V			P	
SWAT	CASE	PROKIT WORKB	MCDONNELL DOUGL	ST LOUIS	MO					P	
SWAT	CASE	PRISM	INDEX TECHNOLOG	CAMBRIDGE	MA					P	
SWAT	CASE	PRIDE	M. BRYCE ASSOC	PALM HARBOR	FL		V				M
SWAT	CASE	POWER TOOLS	ICONIX	SANTA MONICA	CA					P	M
SWAT	CASE	POSE	COMP SYS ADVI	WOODCLIFFLAKE	NJ						
SWAT	CASE	MODEL	COMP COMMAND &	PHILADELPHIA	PA	I	V			P	
SWAT	CASE	MICROSTEP	SYSCORP	AUSTIN	TX						
SWAT	CASE	MATRIX	INTEGRATED SYS	HARTFORD	CT		V	S	A	P	
SWAT	CASE	MACDESIGNER	EXCEL SOFTWARE	MARSHALLTOWN	IA						M
SWAT	CASE	MACBUBBLES	STAR SYS, INC	SILVER SPRING	MD						M
SWAT	CASE	MACANALYST	EXCEL SOFTWARE	MARSHALLTOWN	IA						M
SWAT	CASE	HYPERVIEWMIC	TEN X TECHNOLOG	AUSTIN	TX						
SWAT	CASE	H.P.TEAMWORK	HEWLETT PACKARD	HUNTSVILLE	AL						
SWAT	CASE	GENERATOR II	STRATEGIC ADVAN	TOPEKA	KS					P	
SWAT	CASE	EXCELERATOR	INDEX TECHNOLOG	CAMBRIDGE	MA	I	V	S	A	P	
SWAT	CASE	EXCELER/RTS	INDEX TECHNOLOG	CAMBRIDGE	MA		V	S	A	P	
SWAT	CASE	EPOS	S/W PRODUCTS SE	NEW YORK	NY		V	S	A	P	
SWAT	CASE	EIFFEL	INTERACTIVE SW	GOLETA	CA		V	S	A	P	
SWAT	CASE	DESIGNER	MENTOR GRAPHICS	ORLANDO	FL		V		A		
SWAT	CASE	DESIGN AID	NASTEC	FALLS CHURCH	VA		V			P	
SWAT	CASE	CUSTOMIZER	INDEX TECHNOLOG	CAMBRIDGE	MA					P	
SWAT	CASE	CRADLE	YOURDON INC	MCLEAN	VA						
SWAT	CASE	CARDTOOLS	READY SYSTEMS	PALO ALTO	CA		V			P	
SWAT	CASE	BLUES	ADVANCED LOGICA	BEVERLY HILLS	CA						M
SWAT	CASE	AUTOCODE	INTEGRATED SYS	HARTFORD	CT		V	S	A		
SWAT	CASE	AUDITOR	MENTOR GRAPHICS	ORLANDO	FL		V		A		
SWAT	CASE	ANATOOL	ADVANCED LOGICA	BEVERLY HILLS	CA						M
SWAT	CASE	ANALYST/RT	MENTOR GRAPHICS	ORLANDO	FL		V		A		
SWAT	CASE	ANAL/DES	YOURDON INC	MCLEAN	VA					P	

Technology for the Assessment of Software Quality "TASQ"

APPL_N	TYPE	TOOL_NAME	COMPANY	CITY & STATE		IBM	VAX	SUN	APOLLO	IBM	MAC
SWAT	CODE CONVERT	FORTRIX-ADA	RAPITECH	SUFFERN	NY	I	V			P	
SWAT	CODE CONVERT	COBLIX-C	RAPITECH	SUFFERN	NY						
SWAT	CODEGENERATO	PRO/SOURCE	PROMOD	LAGUNA HILLS	CA		V			P	
SWAT	COMPARATOR	S/COMPARE	ALDON COMPUTER	OAKLAND	CA	I	V				
SWAT	COMPARATOR	EXDIFF	S/W RESEARCH IN	SAN FRANCISCO	CA	I	V	S			
SWAT	COMPARATOR	COMPAREX	A FEW GOOD PEOP	CREST PARK	CA	I					
SWAT	COMPARATOR	COMPARE/HARM	ALDON COMPUTER	OAKLAND	CA	I	V				
SWAT	COMPARATOR	CA-ACCUCHECK	COMP ASSOC INT	GARDEN CITY	NY	I					
SWAT	COMPILER	Z80 C COMPIL	2500 AD S/W	BUEUAVISTA	CA		V			P	
SWAT	COMPILER	Z80 C COMPIL	S/W DEVELOPMENT	DOWNERS GROVE	IL		V	S	A		
SWAT	COMPILER	VAST-2	INTEL SCIENTIFI	BEAVERTON	OR						
SWAT	COMPILER	VADS	VERDIX CORP	CHANTILLY	VA		V	S			
SWAT	COMPILER	TARTAN ADA	TARTAN LAB, INC	PITTSBURGH	PA		V	S			
SWAT	COMPILER	REX-SMA	SYS & S/W INC	COSTA MESA	CA		V	S	A		
SWAT	COMPILER	REX-PL/M/86	SYS & S/W INC	COSTA MESA	CA		V	S	A		
SWAT	COMPILER	REX-C/86	SYS & S/W INC	COSTA MESA	CA		V	S	A	P	
SWAT	COMPILER	PROLOG COMP	ARITY CORP	CONCORD	MA					P	
SWAT	COMPILER	OREGON MOD2	OREGON S/W INC	PORTLAND	OR		V	S			
SWAT	COMPILER	MICRO CONT C	ARCHIMEDES S/W	SAN FRANCISCO	CA		V			P	
SWAT	COMPILER	MAC ADA COMP	MERIDIAN	LAGUNA HILLS	CA					P	
SWAT	COMPILER	JOVIAL COMPI	ADVANCED COMPUT	NEW YORK	NY	I	V				
SWAT	COMPILER	JOVIAL COMP	INTERACT CORP	NEW YORK	NY	I	V				
SWAT	COMPILER	JOV73 COMP	PSS	SANTA MONICA	CA						
SWAT	COMPILER	HP9000 FORTR	HEWLETT PACKARD	HUNTSVILLE	AL						
SWAT	COMPILER	HP-UXC COMPI	HEWLETT PACKARD	HUNTSVILLE	AL						
SWAT	COMPILER	HP-UX PASCAL	HEWLETT PACKARD	HUNTSVILLE	AL						
SWAT	COMPILER	HP-UX ADACOM	HEWLETT PACKARD	HUNTSVILLE	AL						
SWAT	COMPILER	HARRIS ADA C	HARRIS CORP	FT. LAUDERDAL	FL						
SWAT	COMPILER	HAPSE/SE	HARRIS CORP								
SWAT	COMPILER	HAPSE	HARRIS CORP	FT LAUDERDALE	FL						

APPL_N	TYPE	TOOL_NAME	COMPANY	CITY & STATE		IBM	VAX	SUN	APOLLO	IBM	MAC
SWAT	COMPILER	GOULD ADA CO	GOULD, INC.	HUNTSVILLE	AL						
SWAT	COMPILER	FORTRAN COMP	ADVANCED COMPUT	NEW YORK	NY	I	V		A	P	
SWAT	COMPILER	EMCORE COMPI	EMCORE COMP. CO	ATLANTA	GA						
SWAT	COMPILER	DACS	DDC-1 INC	PHOENIX	AZ		V				
SWAT	COMPILER	COMPILERS	LANGUAGE PROCES					S		P	
SWAT	COMPILER	C CROSS COMP	LATTICE INC	LOMBARD	IL					P	
SWAT	COMPILER	APL PLUS	STSC INC	ROCKVILLE	MD	I	V			P	M
SWAT	COMPILER	ADA-86	SOFTECH	WALTHAM	MA		V				
SWAT	COMPILER	ADA TOOL SET	CONVEX COMPUTER	RICHARDSON	TX						
SWAT	COMPILER	ADA COMPILER	CONCURRENT COMP	HUNTSVILLE	AL						
SWAT	COMPILER	ADA COMPILER	ADVANCED COMPUT	NEW YORK	NY		V				
SWAT	COMPILER	ADA COMPILER	ALSYS	WALTHAM	MA	I		S	A	P	M
SWAT	COMPILER	ADA COMPILER	PSS	SANTA MONICA	CA						
SWAT	COMPILER	ADA COMPILER	IRVINE COMPILER	IRVINE	CA		V	S			
SWAT	COMPILER	ADA 1750A CO	INTERACT CORP	NEW YORK	NY	I					
SWAT	COMPILER	88000 COMP	GREEN HILLS S/W	GLENDALE	CA		V	S			
SWAT	COMPILER	68000 COMP	GREEN HILLS S/W	GLENDALE	CA		V	S		P	
SWAT	COMPLIER	OBJECTIVE-C	STEPSTONE CORP	SANDY HOOK	CT		V	S	A	P	
SWAT	COMPLILER	GENEPAK	APPLIED MICROSY	REDMOND	WA		V	S	A	P	
SWAT	CROSS ASSEM	BSW/CROSS AS	BERKELEY SOFTWO	BERKELEY	CA						
SWAT	CROSS ASSEMB	UNIWARE CROS	S/W DEVELOPMENT	DOWNERS GROVE	IL		V	S	A		
SWAT	CROSS ASSEMB	CROSS ASSEMB	COMP SYS CONS	CONYERS	GA						
SWAT	CROSS ASSEMB	CROSS ASSEMB	INTERMETRICS SW	CAMBRIDGE	MA		V	S	A	P	
SWAT	CROSS COMP	C CROSS COMP	INTERMETRICS SW	CAMBRIDGE	MA		V	S	A	P	
SWAT	CROSS COMPIL	PASCAL CROSS	MICROTEC RESEA	SANTA CLARA	CA		V	S	A	P	
SWAT	CROSS COMPIL	HIGH C CROSS	META WARE INC	SANTA CRUZ	CA	I	V	S		P	
SWAT	CROSS COMPIL	FORTAN CROSS	MICROTEC RESEA	SANTA CLARA	CA		V	S	A	P	
SWAT	CROSSCOMPILE	CROSSCOMPILE	MICROTEC RESEA	SANTA CLARA	CA		V	S	A	P	
SWAT	DAT ANALYZER	CAPBAK	S/W RESEARCH IN	SAN FRANCISCO	CA	I	V	S			
SWAT	DDL	BYRON	INTERMETRICS SW	CAMBRIDGE	MA	I	V				

Technology for the Assessment of Software Quality "TASQ"

APPL_N	TYPE	TOOL_NAME	COMPANY	CITY & STATE		IBM	VAX	SUN	APOLLO	IBM	MAC
SWAT	DEBUG	PROBE-68K SY	S/W COMPONENTS	SANTA CLARA	CA						
SWAT	DEBUGGER	XDB	APPLIED MICROSY	REDMOND	WA		V	S	A	P	
SWAT	DEBUGGER	VALISOFT/286	APPLIED MICROSY	REDMOND	WA		V	S	A	P	
SWAT	DEBUGGER	VALID/XEL	APPLIED MICROSY	REDMOND	WA		V	S	A	P	
SWAT	DEBUGGER	VALID/SOFT	APPLIED MICROSY	REDMOND	WA		V	S	A	P	
SWAT	DEBUGGER	UDB	OASYS	WALTHAM	MA		V	S			
SWAT	DEBUGGER	TRACK	PERFORMANCE S/W	RICHMOND	VA	I					
SWAT	DEBUGGER	SYMDUMP	ONLINE S/W INT			I					
SWAT	DEBUGGER	SYMBOLIC DEB	INTERACT CROP	NEW YORK	NY		V	S			
SWAT	DEBUGGER	SIMCASE	ARCHIMEDES S/W	SAN FRANCISCO	CA					P	
SWAT	DEBUGGER	SIM/DEBUGGER	2500 AD S/W	BUEUAVISTA	CA		V			P	
SWAT	DEBUGGER	RT SOURCE	READY SYSTEMS	SUNNYVALE	CA			S			
SWAT	DEBUGGER	RT SCOPE	READY SYSTEMS	PALOALTO	CA		V				
SWAT	DEBUGGER	REX-SOFTPROB	SYS & S/W INC	COSTA MESA	CA		V	S	A	P	
SWAT	DEBUGGER	INTERTEST	ONLINE S/W INT	CALVERTON	MD	I					
SWAT	DEBUGGER	GENEPROBE	APPLIED MICROSY	REDMOND	WA		V	S	A	P	
SWAT	DEBUGGER	FDEBUG	OASYS	WALTHAM	MA		V	S	A		
SWAT	DEBUGGER	EMUPAK	APPLIED MICROSY	REDMOND	WA		V	S	A	P	
SWAT	DEBUGGER	DEBUGGER-SIM	MICROTEC RESEA	SANTA CLARA	CA		V	S	A	P	
SWAT	DEBUGGER	CSR BUG BYTE	COMP SYS RESEAR	AVON	CT	I					
SWAT	DEBUGGER	CROSS DEBUG	HEWLETT PACKARD	HUNTSVILLE	AL						
SWAT	DEBUGGER	CONCUR DEBUG	INTEL SCIENTIFI	BEAVERTON	OR						
SWAT	DEBUGGER	CICS-dBUG-AI	COMPUWARE CORP	BIRMINGHAM	MI	I					
SWAT	DEBUGGER	CICS RADAR	COMPUWARE CORP	BIRMINGHAM	MI	I					
SWAT	DEBUGGER	CICS ABEND-A	COMPUWARE CORP	BIRMINGHAM	MI	I					
SWAT	DEBUGGER	CDEBUG	OASYS	WALTHAM	MA		V	S	A		
SWAT	DEBUGGER	CDB	THIRD EYE S/W	MENLO PARK	CA	I	V	S	A	P	
SWAT	DEBUGGER	ARTSCOPE6802	READY SYSTEMS	SUNNYVALE	CA		V				
SWAT	DEBUGGER	AIEM	S/W SYS DESIGN	CLAREMONT	CA		V	S	A	P	
SWAT	DESIGN	COMPOSER	XINOTECH RESEAR	MINNEAPOLIS	MN		V			P	

Technology for the Assessment of Software Quality "TASQ"

APPL_N	TYPE	TOOL_NAME	COMPANY	CITY & STATE		IBM	VAX	SUN	APOLLO	IBM	MAC
SWAT	DESIGN LANG.	POWER PDL	ICONIX	SANTA MONICA	CA		V			P	M
SWAT	DESIGNLANGPR	RE-SPEC	S/W PRODUCTS SE	NEW YORK	NY		V	S	A	P	
SWAT	DEV.ENVIRON	SYS BUNDLES	HEWLETT PACKARD	HUNTSVILLE	AL						
SWAT	DISASSEMBLER	680X/6502 SU	COMP SYS CONS	CONYERS	GA						
SWAT	DOC FORMAN	PHILE-68K FI	S/W COMPONENTS	SANTA CLARA	CA						
SWAT	DOC FORMAT	DOC TEMPLATE	INTERMETRICS SW	CAMBRIDGE	MA		V	S	A	P	
SWAT	DOC FORMAT	DOC GEN	INTERMETRICS SW	CAMBRIDGE	MA		V	S	A	P	
SWAT	DOC FORMATER	BUILD	LEVERAGE S/W IN							P	
SWAT	DRIVER	VALID/ES DRI	APPLIED MICROSY	REDMOND	WA		V	S	A	P	
SWAT	DYNAM SIM	TPNS	IBM	HUNTSVILLE	AL	I					
SWAT	DYNAM SIM	SIMULATOR	OASYS	WALTHAM	MA		V	S	A		
SWAT	DYNAM SIM	IPSC SIMULAT	INTEL SCIENTIFI	BEAVERRTON	OR		V	S			
SWAT	DYNAM SIM	BATCH TERMIN	IBM	HUNTSVILLE	AL	I					
SWAT	DYNAM. SIM	1750 ISA SIM	INTERACT CORP	NEW YORK	NY		V	S			
SWAT	DYNAMIC	TRACE	AK, INC.	SAN JOSE	CA	I				P	
SWAT	DYNAMIC	TESTGEN	S/W SYS DESIGN	CLAREMONT	CA		V	S	A	P	
SWAT	DYNAMIC	TCAT	S/W RESEARCH IN	SAN FRANCISCO	CA		V	S	A	P	
SWAT	DYNAMIC	S-TCAT/C	S/W RESEARCH IN	SAN FRANCISCO	CA		V	S	A	P	
SWAT	DYNAMIC	PSOS-68K SYS	S/W COMPONENTS	SANTA CLARA	CA						
SWAT	DYNAMIC	PCA	DEC	NASHUA	NH		V				
SWAT	DYNAMIC	J73AVS	WRIGHT PAT AFB	DAYTON	OH	I	V				
SWAT	DYNAMIC	HI LEV. SW	HEWLETT PACKARD	HUNTSVILLE	AL						
SWAT	DYNAMIC	FPE	SOFTOOL CORPORA	GOLETA	CA		V				
SWAT	DYNAMIC	FACES	COSMIC			I					
SWAT	DYNAMIC	CPE	SOFTOOL CORPORA	GOLETA	CA		V				
SWAT	DYNAMIC	C-PE	SOFTOOL CORPORA	GOLETA	CA		V				
SWAT	DYNAMIC SIM	XCELL +	PRITSKER & ASSO							P	
SWAT	DYNAMIC SIM	TESS	PRITSKER & ASSO							P	
SWAT	DYNAMIC SIM	SLAM II	PRITSKER & ASSO	WEST LAFAYETT	IN					P	
SWAT	ENG. SIM	NEKTON	INTEL SCIENTIFI	BEAVERTON	OR						

Technology for the Assessment of Software Quality "TASQ"

APPL_N	TYPE	TOOL_NAME	COMPANY	CITY & STATE		IBM	VAX	SUN	APOLLO	IBM	MAC
SWAT	ENVIRON LANG	PASCAL-2	OREGON S/W INC	PORTLAND	OR		V	S			
SWAT	ENVIRONMENT	NAG FORTRAN	NUMERICAL ALGON	DOWNERS GROVE	IL		V	S		P	
SWAT	ENVIRONMENT	MODULA2/68	ANA SYSTEMS	FOSTER CITY	CA			S	A		
SWAT	ENVIRONMENT	MOD2/68-CD	ANA SYSTEMS	FOSTER CITY	CA		V				
SWAT	ENVIRONMENT	ADV TOOL KIT	ARITY CORP	CONCORD	MA					P	
SWAT	FLOW CHART	PROMOD/DC	PROMOD	LAGUNA HILLS	CA		V			P	
SWAT	FLOW CHART	EASYFLOW	HAVENTREE	THOUSAND ISL.	NY					P	
SWAT	FLOW CHARTER	JCLFLOW	CONSUMER SYSTEM	LOMBARD	IL	I					
SWAT	FLOWCHART	PASSAGE	INTEL SCIENTIFI	BEAVERTON	OR						
SWAT	FLOWCHART	FREEFLOW	ICONIX	SANTA MONICA	CA						M
SWAT	FLOWCHART	CLEAR +	CLEAR SOFTWARE	BROOKLINE	MA	I					
SWAT	HW STIMULAT	ECHOVAX	ARIUM	ANAHEIM	CA		V				
SWAT	HW TEST EQUI	ASIC ANALYZE	GOULD INC DESIG	CUPERTINO	CA						
SWAT	INTERPRETER	SAFE C INTER	CATALYTIX	CAMBRIDGE	MA		V	S	A		
SWAT	IPSE	MAESTRO	SOFTLAB INC	CHICAGO	IL	I					
SWAT	LANGUAGE	UNH PROLOG	UNIV OF NH	DURHAM	NH		V			P	
SWAT	LANGUAGE	METASTEP ASS	STEP ENGINEERIN	SUNNYVALE	CA						
SWAT	LIBRARY	PROGRAM LIB	INTERMETRICS SW	CAMBRIDGE	MA		V	S	A	P	
SWAT	LIBRARY	LIBRRARIAN	S/W DEVELOPMENT	DOWNERS GROVE	IL		V	S	A		
SWAT	LIBRARY	LIBRARIAN	MICROTEC RESEA	SANTA CLARA	CA		V	S	A	P	
SWAT	LIBRARY	IPSC-VX VECL	INTEL SCIENTIFI	BEAVERTON	OR						
SWAT	LIBRARY	INTERWORK II	INTEL SCIENTIFI	BEAVERTON	OR						
SWAT	LIBRARY	GRACE	EVB S/W ENG INC	FREDERICK	MD	I	V	S	A	P	M
SWAT	LIBRARY	CEPHES	OASYS	WALTHAM	MA						
SWAT	LINKER	LINKER	S/W DEVELOPMENT	DOWNERS GROVE	IL		V	S	A		
SWAT	LINKER	LINKER	2500 AD S/W	BUEUAVISTA	CA		V	S	A	P	
SWAT	LINKER	GENELINK	APPLIED MICROSY	REDMOND	WA		V	S	A	P	
SWAT	LINKER	1750A LINK	PSS	SANTA MONICA	CA						
SWAT	LINKER/ASSEM	OL/A	OREGON S/W INC	PORTLAND	OR	I	V	S			
SWAT	LINKERS	LINKING LOAD	MICROTEC RESEA	SANTA CLARA	CA		V	S	A	P	

Technology for the Assessment of Software Quality "TASQ"

APPL_N	TYPE	TOOL_NAME	COMPANY	CITY & STATE		IBM	VAX	SUN	APOLLO	IBM	MAC
SWAT	MACRO	MACROS	2500 AD S/W	BUEUAVISTA	CA						
SWAT	METHODOLOGY	SADT	SOFTECH	WALTHAM	MA						
SWAT	MICRO PROCES	UNIWARE 6800	S/W DEVELOPMENT	DOWNERS GROVE	IL		V	S	A		
SWAT	OS	HP-UX OS 300	HEWLETT PACKARD	HUNTSVILLE	AL						
SWAT	OS	HP-UX OS 200	HEWLETT PACKARD	HUNTSVILLE	AL						
SWAT	PATH ANALYZ	PAT	SAIC	ARLINGTON	VA	I	V		A		
SWAT	PATH ANALYZE	BASIS BRANCH	HEWLETT PACKARD	HUNTSVILLE	AL						
SWAT	POST EX. ANA	TRAPS	TRAV TECH, INC.	NEWMAN	GA	I	V				
SWAT	POST EXECUTI	EXTRACT	ALDON COMPUTER	OAKLAND	CA						
SWAT	PROGRESSTEST	AUTOTESTER	S/W RECORDING	DALLAS	TX					P	
SWAT	PROJ. ENVIRO	HP-UX TOOLS	HEWLETT PACKARD	HUNTSVILLE	AL						
SWAT	PUBLICATION	DESIGN BOOK	TASC	ARLINGTON	VA						
SWAT	QUALITY METR	DQM	HUGHES			I					
SWAT	QUALITY METR	CMT	EVB S/W ENG INC	FREDERICK	MD	I	V	S	A	P	M
SWAT	QUALITY METR	BAT/DAT	MCCABE & ASSOC	COLUMBIA	MD					P	
SWAT	QUALITY METR	ACT/CAT	MCCABE & ASSOC	COLUMBIA	MD					P	
SWAT	REG TRACE	RVTS	TELEDYNE BROWN	HUNTSVILLE	AL					P	
SWAT	REG TRACE	RTRACE	NASTEC	FALLS	VA						
SWAT	REGRESSION T	TEST/IMS	CONSUMER SYSTEM	LOMBARD	IL						
SWAT	REGRESSION T	TEST MANAGER	DEC	NASHUA	NH		V				
SWAT	REGRESSION T	SMARTS	S/W RESEARCH IN	SAN FRANCISCO	CA		V			P	
SWAT	REGRESSION T	IPAT	INTEL CORPORATI	SANTA CLARA	CA					P	
SWAT	REGRESSION T	FASTTASK	ICONIX	SANTA MONICA	CA						M
SWAT	REGRESSION T	CICS PLAYBAC	COMPUWARE CORP.	BIRMINGHAM	MI	I					
SWAT	REGRESSON TE	S/W PERFORMA	HEWLETT PACKARD	HUNTSVILLE	AL						
SWAT	REPRESSION T	DCATS	SYS DES & DEV	BOULDER	CO					P	
SWAT	REQ TRACE	THOR	SAIC	HUNTSVILLE	AL						
SWAT	REQ TRACE	RT2	BOEING	SEATTLE	WA						
SWAT	REQ TRACE	REX	LEVERAGE S/W IN	ALEXANDRIA	VA					P	
SWAT	REQ TRACE	RETT	S/W SYS DESIGN	CLAREMONT	CA		V	S	A	P	

Technology for the Assessment of Software Quality "TASQ"

APPL_N	TYPE	TOOL_NAME	COMPANY	CITY & STATE		IBM	VAX	SUN	APOLLO	IBM	MAC
SWAT	REQ TRACE	PROCAP/TMS	PROMOD	LAGUNA HILLS	CA		V			P	
SWAT	REQ TRACE	ARTS	LOCKHEED								
SWAT	REQ. ANALYZE	ARIS	S/W SYS DESIGN	CLAREMONT	CA		V	S	A	P	
SWAT	REQ. TRACE	TURBO TRACE	COMP SCI INNOV	PALM BAY	FL		V				
SWAT	RESTRUC.PROG	SMARTCHART	ICONIX	SANTA MONICA	CA						M
SWAT	RUNTIME	STANDALONE A	SAIC	HUNTSVILLE	AL						
SWAT	S/W TEST H/W	SPA	HEWLETT PACKARD	HUNTSVILLE	AL						
SWAT	SIM LANGUAGE	SIMSCRIPT II	CACI INC	LA JOLLA	CA	I				P	
SWAT	SIMULATOR	SIMULATOR	COMP SYS CONS	CONYERS	GA						
SWAT	SOURCE CONVE	CA-CONVERTOR	COMP ASSOC INT	GARDEN CITY	NY	I					
SWAT	STANDARDS CO	JCLAUDIT	CONSUMER SYSTEM	LOMBARD	IL	I					
SWAT	STATIC	SCOREBOARD	TRAV TECH INC	NEWMAN	GA	I				P	
SWAT	STATIC	SCA	DEC	NASHUA	NH		V				
SWAT	STATIC	SAP	COSMIC				V				
SWAT	STATIC	SAFE C RUNTI	CATALYTIX	CAMBRIDGE	MA		V	S	A		
SWAT	STATIC	RXVP80	GEN RESRCH CORP	SANTA BARBARA	CA		V				
SWAT	STATIC	RA	FORD								
SWAT	STATIC	PROCAP	PROMOD	LAGUNA HILLS	CA		V			P	
SWAT	STATIC	MSAT	ELECTRONIC PROV	FT. HUACHUCA			V				
SWAT	STATIC	MAT	SAIC	ARLINGTON	VA	I	V				
SWAT	STATIC	LOGISCOPE	VERILOG USA	ALEXANDRIA	VA		V	S	A		
SWAT	STATIC	LASE	LITTON				V				
SWAT	STATIC	ISAS	SINGER	TUSCON	AZ		V				
SWAT	STATIC	GYPSY VERIFI	COMPUTATIONALOG	AUSTIN	TX			S			
SWAT	STATIC	AMS	HARRIS CORP	MELBOURNE	FL		V				
SWAT	STATIC	ADF	M. BRYCE ASSOC	PALM HARBOR	FL		V				
SWAT	STATIC	ADAMAT	DYNAMICS RESEAR	ANDOVER	MA		V	S			
SWAT	STATIC	ADADL	S/W SYS DESIGN	CLAREMONT	CA		V	S	A	P	
SWAT	STRESS TEST	VERIFY	ONLINE S/W INT	CALVERTON	MD	I					
SWAT	STRESS TEST	AUTOMATOR MI	INTERACTIVE SOL	BOGOTA	NJ	I				P	

Technology for the Assessment of Software Quality "TASQ"

APPL_N	TYPE	TOOL_NAME	COMPANY	CITY & STATE		IBM	VAX	SUN	APOLLO	IBM	MAC
SWAT	SYS SIMULATI	PAWS	INFO RESEARCH A	AUSTIN	TX	I	V	S	A		
SWAT	TEST GENERAT	CA-DATAMACS	COMP ASSOC INT	GARDEN CITY	NY	I					
SWAT	TRANSLATOR	TRANSLATOR	COMP SYS CONS	CONYERS	GA						
SWAT	TRANSLATOR	TRANSLATOR	2500 AD S/W	BUEUAVISTA	CA					P	
SWAT	TRANSLATOR	FACT	SOFTOOL CORPORA	GOLETA	CA		V				
SWAT	TRANSLATOR	ENG TO C/C T	CATALYTIX	CAMBRIDGE	MA		V	S	A		
SWAT	UNDEVELOPED	STAT	AMCCOM	PARSIPPANY	NJ						
SWAT	UNDEVELOPED	SOFTCAT	ATAC	MOUNTAINVIEW	CA	I					
SWAT	UNDEVELOPED	ESRA	AMCCOM	PARSIPPANY	NJ						
SWAT	UNDEVELOPED	DES COMPL ME	AMCCOM	PARSIPPANY	NJ						
SWAT	UTILITY	SNDRCV	OASYS	WALTHAM	MA		V	S	A	P	
SWAT	UTILITY	OBJMODCONVER	OASYS	WALTHAM	MA		V	S	A		
SWAT	X-COMPILER	CROSS COMPIL	ALSYS	WALTHAM	MA		V	S		P	
PSM		XED MENU PRO	COMPUTER METHOD	CHATSWORTH	CA						
PSM	ACU. OPTION	WBS	COMP COGNITION	SAN DIEGO	CA	I	V	S	A		
PSM	ACU. OPTION	PO&R	COMP COGNITION	SAN DIEGO	CA	I	V	S	A		
PSM	ACU. OPTION	PAYROLL	COMP COGNITION	SAN DIEGO	CA	I	V	S	A		
PSM	ACU. OPTION	ACCOUNTS PAY	COMP COGNITION	SAN DIEGO	CA	I	V	S	A		
PSM	CASE	PROJECT SOFT	ATHERTON TECHNO	SUNNYVALE	CA		V	S			
PSM	CONFIG MGT	TEAM VISION	TEAM 1 SYS INC	SUNNYVALE	CA	I	V	S	A	P	
PSM	CONFIG MGT	SCONS	CORP COMP SYS	HOLMDEL	NJ						
PSM	CONFIG MGT	PVCS	POLYTRON	BEAVERTON	OR		V			P	
PSM	CONFIG MGT	PROMOD CM	PROMOD	LAKE FOREST	CA		V				
PSM	CONFIG MGT	POLYMAKE	POLYTRON	BEAVERTON	OR		V			P	
PSM	CONFIG MGT	ENDEVOR C1	BST	WESTBORO	MA	I					
PSM	CONFIG MGT	CUE	K&H PROJ SYS IN	SPARTA	NJ	I	V			P	
PSM	CONFIG MGT	CMT	EXPERTWARE INC	SANTA CLARA	CA	I	V	S	A	P	M
PSM	CONFIG MGT	CCC	SOFTOOL CORP	GOLETA	CA		V	S		P	
PSM	CONFIG MGT	AIDE-DE-CAMP	S/W MAIN. & DEV	CONCORD	MA		V	S	A		
PSM	CONFIG. MGMT	DELTA	CORPORATE COMPU	HOLMDEL	NJ		V				

Technology for the Assessment of Software Quality "TASQ"

APPL_N	TYPE	TOOL_NAME	COMPANY	CITY & STATE		IBM	VAX	SUN	APOLLO	IBM	MAC
PSM	COST MODEL	SPACE	CECOM PA&T	FT MONMOUTH	NJ					P	
PSM	COST MODEL	SOFTQUAL	CARMAN GROUP	PLANO	TX					P	
PSM	COST MODEL	SOFTCOST-R	REIFER CONSULT.	TORRANCE	CA					P	
PSM	COST MODEL	SOFTCOST ADA	REIFER CONSULT.	TORRANCE	CA					P	
PSM	COST MODEL	SLIM	QUANTITATIVE SW	MCLEAN	VA					P	
PSM	COST MODEL	SIZE PLANNER	QUANTITATIVE SW	MCLEAN	VA					P	
PSM	COST MODEL	SECOMO	TELOS FED SYS	SHREWSBURY	NJ					P	
PSM	COST MODEL	P3	QUANTITATIVE SW	MCLEAN	VA					P	
PSM	COST MODEL	COCOPRO	ICONIX	SANTA MONICA	CA						M
PSM	COST MODEL	COCOMO	TRW-DSG	RENDONDO BEAC	CA						
PSM	COST MODEL	ASSET-R	REIFER CONSULT.	TORRANCE	CA					P	
PSM	DOCUMENT MGT	SLATE	BBN S/W PRODUCT	CAMBRIDGE	MA			S			
PSM	DOCUMENT MGT	INTERLEAF TP	INTERLEAF			I	V	S	A	P	M
PSM	DOCUMENT MGT	DST	EXPERTWARE INC	SANTA CLARA	CA	I	V	S	A	P	
PSM	DOCUMENT MGT	CRYSTAL DMS	SYNTACTICS	SANTA CLARA	CA			S	A	P	
PSM	DOCUMENT MGT	CONCORDIA	SYMBOLICS	CAMBRIDGE	MA						M
PSM	DOCUMENT MGT	4TH-WRITE	ADVENTUREWARE	ALLISON PARK	PA	I	V			P	
PSM	EDITOR	VI-PLUS	UNIPRESS S/W	EDISON	NJ	I	V	S	A	P	
PSM	EDITOR	KEYONE EDITO	OASYS	WALTHAM	MA		V	S		P	
PSM	EDITOR	EMACS	UNIPRESS S/W	EDISON	NJ	I	V	S	A	P	
PSM	EDITOR	CEPAGE	S/W ENGIN INC.			I	V	S	A	P	
PSM	EDITOR	C-MACS	UNIPRESS S/W	EDISON	NJ	I	V	S	A	P	
PSM	EDITOR	APOGEE EDIT	BINARY RESEARCH	FT WASHINGTON	PA	I	V			P	
PSM	EDITOR	ADADL EDITOR	S/W SYS DE	CLAREMONT	CA	I	V	S	A	P	
PSM	GRAPHICS PK	TELEGRAF	COMPUTER ASSOC.	ATLANTA	GA	I	V				
PSM	GRAPHICS PK	TAB GEN OPTI	CONTEXT	BEAVERTON	OR				A		
PSM	GRAPHICS PK	STATGRAPHICS	STSC	ROCKVILLE	MD					P	
PSM	GRAPHICS PK	SCDDRAW	MCDONNELL DOUGL							P	
PSM	GRAPHICS PK	RENDOR	MULTIWARE INC	DAVIS	CA		V				
PSM	GRAPHICS PK	QUAL GRAPHIC	QUALITY S/W PRO	BEVERLY HILLS	CA	I	V	S	A	P	

APPL_N	TYPE	TOOL_NAME	COMPANY	CITY & STATE		IBM	VAX	SUN	APOLLO	IBM	MAC
PSM	GRAPHICS PK	Q-CHART	QUADRATRON SYS	SHERMAN OAKS	CA	I	V	S	A	P	
PSM	GRAPHICS PK	PICSURE	PRECISIONVISUAL	BOULDER	CO	I	V	S	A		
PSM	GRAPHICS PK	PICSURE	PRECISIONVISUAL	BOULDER	CO	I	V	S	A		
PSM	GRAPHICS PK	PICED EDITOR	CONTEXT	BEAVERTON	OR				A		
PSM	GRAPHICS PK	PGB 300	UNIPRESS S/W	EDISON	NJ	I	V	S	A	P	
PSM	GRAPHICS PK	MGSP	MULTIWARE INC	DAVIS	CA		V				
PSM	GRAPHICS PK	MCPAINTII	CLARIS	MOUNTAIN VIEW	CA						M
PSM	GRAPHICS PK	MCDRAW II	CLARIS	MOUNTAIN VIEW	CA						M
PSM	GRAPHICS PK	IPT GRAFSMAN	SOUTHWIND S/W	WICHATA	KS		V	S	A	P	
PSM	GRAPHICS PK	GRAPHIC GATE	CONTEXT	BEAVERTON	OR				A		
PSM	GRAPHICS PK	GRAFX GRAPHI	D&E S/W INC	CLINTON	MD		V				
PSM	GRAPHICS PK	GRAFMAKER	PRECISIONVISUAL	BOULDER	CO	I	V	S	A		
PSM	GRAPHICS PK	EZGRAF	SOUTHWIND S/W	WICHITA	KS		V	S	A	P	
PSM	GRAPHICS PK	DOC	CONTEXT	BEAVERTON	OR				A		
PSM	GRAPHICS PK	DI 3000	PRECISIONVISUAL	BOULDER	CO	I	V	S	A		
PSM	GRAPHICS PK	DFDDRAW	MCDONNELL DOUGL							P	
PSM	GRAPHICS PK	DESIGN	META SOFTWARE	CAMBRIDGE	MA	I		S		P	M
PSM	GRAPHICS PK	CUSTOMGANTT	SOFTRAK SYS	SALT LAKE CTY	UT		V	S		P	
PSM	GRAPHICS PK	ARTISAN	MEDIA LOGIC INC	SANTA MONICA	CA						
PSM	METHODODLOGY	STRADIS	MCDONNELL DOUGL								
PSM	MGT TRAINER	GREMEX	COSMIC	ATHENS	GA	I					
PSM	OFFICE AUTO.	SMARTWARE	INFORMIX S/W	LENEXA	KS			S	A	P	
PSM	PERF MEA SYS	I/CSCS	K&H PROJ SYS IN	SPARTA	NJ		V				
PSM	PROB TRACKER	STR TRACKER	AMCCOM	PICATINNY ARS	NJ		V				
PSM	PROJ MANAGER	WINGS	AGS MGT. SYS.	KING OF PRUSS	PA	I	V			P	
PSM	PROJ MANAGER	VUE	NIS, INC.	SAN JOSE	CA	I	V	S	A		
PSM	PROJ MANAGER	TIME LINE	SYMANTEC	CUPERTINO	CA					P	
PSM	PROJ MANAGER	TELLAPLAN	COMPUTER ASSOC.	ATLANTA	GA	I	V	S	A		
PSM	PROJ MANAGER	SUPERPROJECT	COMPUTER ASSOC.	SAN JOSE	CA					P	
PSM	PROJ MANAGER	QWIKNET	PSDI	VALLY FORGE	PA					P	

Technology for the Assessment of Software Quality "TASQ"

APPL_N	TYPE	TOOL_NAME	COMPANY	CITY & STATE		IBM	VAX	SUN	APOLLO	IBM	MAC
PSM	PROJ MANAGER	QUICK-PLAN	MITCHELL MGT	WESTBOROUGH	MA					P	
PSM	PROJ MANAGER	PROJECT/2	PSDI	VALLY FORGE	PA		V				
PSM	PROJ MANAGER	PROJECT WORK	APPLIED BUS TEC	NEW YORK	NY					P	
PSM	PROJ MANAGER	PROJECT PLAN	PRIMAVERA								
PSM	PROJ MANAGER	PRIDE-PSM	M. BRYCE & ASSO	PALM HARBOR	FL	I	V	S	A		
PSM	PROJ MANAGER	PRESTIGE PC	K&H PROJ SYS IN	SPARTA	NJ					P	
PSM	PROJ MANAGER	PREMIS	K&H PROJ SYS IN	WAYNE	PA		V				
PSM	PROJ MANAGER	POWER	EXPERTWARE INC	SANTA CLARA	CA					P	
PSM	PROJ MANAGER	PLOTTRAK	SOFTRAK SYS	SALT LAKE CTY	UT		V	S		P	
PSM	PROJ MANAGER	PLANNER	PROD. SOL., INC	WALTHAM	MA		V				
PSM	PROJ MANAGER	PERT TIME II	COSMIC	ATHENS	GA	I					
PSM	PROJ MANAGER	PC-MAPPS	MITCHELL MGT	WESTBOROUGH	MA					P	
PSM	PROJ MANAGER	PAC MICRO	AGS MGT. SYS.	KING OF PRUSS	PA					P	
PSM	PROJ MANAGER	PAC III	AGS MGT. SYS.	KING OF PRUSS	PA	I	V				
PSM	PROJ MANAGER	NIPS	COSMIC	ATHENS	GA						
PSM	PROJ MANAGER	MULTIPROJ-P	TECHNISOFT	LEVENDIA	CA					P	
PSM	PROJ MANAGER	MULTIPROD-O	TECHNISOFT	LEVENDIA	CA					P	
PSM	PROJ MANAGER	MIS	COSMIC	ATHENS	GA		V				
PSM	PROJ MANAGER	MICROTRAK	SOFTRAK SYS	SALT LAKE CTY	UT		V	S		P	
PSM	PROJ MANAGER	MICROPLANNER	MICRO PLAN INT.	SAN FRANCISCO	CA					P	M
PSM	PROJ MANAGER	MICROPLAN 6	MICRO PLAN INT.	SAN FRANCISCO	CA					P	
PSM	PROJ MANAGER	MICRO-MAPPS	MITCHELL MGT	WESTBOROUGH	MA	I					
PSM	PROJ MANAGER	MASTERPLAN	QUALITY S/W PRO	BEVERLY HILLS	CA	I	V	S	A	P	
PSM	PROJ MANAGER	MASTERPLAN	UNIPRESS S/W	EDISON	NJ	I	V	S	A	P	
PSM	PROJ MANAGER	MAPPS-AI	MITCHELL MGT	WESTBOROUGH	MA	I					
PSM	PROJ MANAGER	MAPPS	MITCHELL MGT.	WESTBOROUGH	MA	I	V				
PSM	PROJ MANAGER	MACPROJECT	CLARIS	MOUNTAIN VIEW	CA						M
PSM	PROJ MANAGER	INSTAPLAN	INSTAPLAN								
PSM	PROJ MANAGER	HARVARD PM30	S/W PUBLISHING	ATLANTA	GA					P	
PSM	PROJ MANAGER	FORESIGHT	HILLORY S/W/ IN	BRIELLE	NJ				A	P	

Technology for the Assessment of Software Quality "TASQ"

APPL_N	TYPE	TOOL_NAME	COMPANY	CITY & STATE		IBM	VAX	SUN	APOLLO	IBM	MAC
PSM	PROJ MANAGER	EASYTRAK	DIGITAL PLANNER	NEWPORT BEACH	CA	I	V	S	A		M
PSM	PROJ MANAGER	EASY PROJECT	TECHNISOFT	LEVENDIA	CA					P	
PSM	PROJ MANAGER	C-PLAN	DSD CORP.	BOTHELL	WA		V			P	
PSM	PROJ MANAGER	ASAPMS	ASA	NEW YORK	NY	I	V	S	A	P	
PSM	PROJ MANAGER	ARTEMIS PC	METIER	HOUSTON	TX					P	
PSM	PROJ MANAGER	ARTEMIS	METIER	HOUSTON	TX						
PSM	PROJ MANAGER	AEPEX	TIMBERLINE S/W	BEAVERTON	OR					P	
PSM	PROJ MANAGER	ACUITY LINE	COMP COGNITION	SAN DIEGO	CA	I	V	S	A		
PSM	PROJ SCHEDUL	SET	COSMIC	ATHENS	GA	I					
PSM	PROJ SCHEDUL	PROJECT	MICROSOFT CORP	REDMOND	WA			S		P	
PSM	PROJ SCHEDUL	PPARS	COSMIC	ATHENS	GA						
PSM	PROJ SCHEDUL	PACE/PRRA	COSMIC	ATHENS	GA						
PSM	PROJ SCHEDUL	PACE	COSMIC	ATHENS	GA						
PSM	PROJ SCHEDUL	MARS	COSMIC	ATHENS	GA		V				
PSM	PROJ SCHEDUL	FMAP	COSMIC	ATHENS	GA	I					
PSM	PROJ SCHELUL	PROJ SCHEDUL	GTSI	CHANTILLY	VA					P	
PSM	PROJ TRACKER	PROTRACS	APPLIED MICROSY	ROSWELL	GA					P	
PSM	PUBLICATION	PM HANDBOOK	APPLIED INFO DE	OAKBROOK	IL						
PSM	QUAL METRICS	ANALYZE	AUTOMETRIC	LINTHICUM	MD	I	V	S	A	P	
PSM	SCREEN MANAG	FORMIX	MCSI	EDEN PRAIRIE	MN	I	V	S	A	P	
PSM	SECURITY	OMNIGUARD	ONLINE S/W INT	CALVERTON	MD	I					
PSM	SOW ASSIST	SOW ASSIST	BELVOIR RD&E	FT BELVOIR	VA					P	
PSM	SPREADSHEET	WINGZ	INFORMIX S/W	LENEXA	KS						M
PSM	SPREADSHEET	TACTICIAN	SOUTHWIND S/W	WICHITA	KS		V	S	A	P	
PSM	SPREADSHEET	SYMPHONY	LOTUS DEVEL COR							P	
PSM	SPREADSHEET	SUPERCALC5	COMPUTER ASSOC.	SAN JOSE	CA					P	
PSM	SPREADSHEET	SCO PROF	UNIPRESS S/W	EDISON	NJ	I	V	S	A	P	
PSM	SPREADSHEET	Q-PLAN	QUADRATRON SYS	SHERMAN OAKS	CA	I	V	S	A	P	
PSM	SPREADSHEET	Q-CALC	QUALITY S/W PRO	BEVERLY HILLS	CA	I	V	S	A	P	
PSM	SPREADSHEET	Q-CALC	UNIPRESS S/W	EDISON	NJ	I	V	S	A	P	

Technology for the Assessment of Software Quality "TASQ"

APPL_N	TYPE	TOOL_NAME	COMPANY	CITY & STATE		IBM	VAX	SUN	APOLLO	IBM	MAC
PSM	SPREADSHEET	PILOT EIS	PILOT	BOSTON	MA		V				
PSM	SPREADSHEET	MULTIPLAN	MICROSOFT CORP	REDMOND	WA					P	
PSM	SPREADSHEET	LOTUS 1-2-3	LOTUS DEVEL COR							P	
PSM	SPREADSHEET	EXCEL	MICROSOFT CORP	REDMOND	WA						
PSM	SPREADSHEET	CHART	MICROSOFT CORP	REDMOND	WA			S		P	
PSM	SPREADSHEET	20/20	ACCESS TECH	SOUTH NATICK	MA	I	V	S		P	
PSM	SQA EXPERT	SQAM	AMCCOM	PICATINNY ARS	NJ		V				
PSM	TRAN UTILITY	PLANLINKS	COMPUTER ASSOC.	SAN JOSE	CA	I	V	S	A		
PSM	TRAN UTILITY	DCA FILTER	CONTEXT	BEAVERTON	OR				A		
PSM	VUE OPTION	RESOUCE LEVE	NIS, INC.	SAN JOSE	CA	I	V	S	A		
PSM	VUE OPTION	PLOTTER GRAP	NIS, INC.	SAN JOSE	CA	I	V	S	A		
PSM	VUE OPTION	MULTI PROJ A	NIS, INC.	SAN JOSE	CA	I	V	S	A		
PSM	VUE OPTION	COST MODULE	NIS, INC.	SAN JOSE	CA	I	V	S	A		
PSM	WORDPROCESSR	XED	COMPUTER METHOD	CHATSWORTH	CA		V	S		P	
PSM	WORDPROCESSR	WORKBENCH	ADDISON-WESLEY	READING	MA					P	M
PSM	WORDPROCESSR	SCOLYRIX	UNIPRESS S/W	EDISON	NJ	I	V	S	A	P	
PSM	WORDPROCESSR	Q-TYPESET	QUADRATRON SYS	SHERMAN OAKS	CA	I	V	S	A	P	
PSM	WORDPROCESSR	Q-ONE	QUADRATRON SYS	SHERMAN OAKS	CA	I	V	S	A	P	
PSM	WORDPROCESSR	MACWRITE	CLARIS	MOUNTAIN VIEW	CA						M
PSM	WORKSTATION	ENGIN WRITER	CONTEXT	BEAVERTON	OR				A		
PSM	WORKSTATION	ENGI WRITER+	CONTEXT	BEAVERTON	OR				A		
PSM	WORKSTATION	CONTEXT WRIT	CONTEXT	BEAVERTON	OR				A		
PSM	WORKSTATION	CONTEXT EDIT	CONTEXT	BEAVERTON	OR				A		
OLEA	ESR	XCON	DEC	HUNTSVILLE	AL		V				
OLEA	ESR	VS ACCELERAT	COYNE KALA INC	ARLINGTON	VA						
OLEA	ESR	TIMM TUNER	GEN RESRCH CORP	MCLEAN	VA		V				
OLEA	ESR	TEST BENCH	CARNEGIE GROUP	PITTSBURGH	PA					P	
OLEA	ESR	PLANNING WS	KNOWLEDGEWARE	ATLANTA	GA					P	
OLEA	ESR	GENSIS	HELP/38 SYSTEMS	MINNETONKA	MN	I					
OLEA	ESR	GEN SIMULAT	PREDICTION SYS								

Technology for the Assessment of Software Quality "TASQ"

APPL_N	TYPE	TOOL_NAME	COMPANY	CITY	& STATE	IBM	VAX	SUN	APOLLO	IBM	MAC
OLEA	ESR	EXTEND	SONICS ENTERPR								
OLEA	ESR	DESIGN WSTAT	KNOWLEDGEWARE	ATLANTA	GA					P	
OLEA	ESR	DESIGN GENER	COMPUTER SCIENC								
OLEA	ESR	CONCEPT MODU	WISDOM SYSTEM	CHAGRM FALLS	OH			S			
OLEA	ESR	AXLE	GOLDHILL COMPUT	CAMBRIDGE	MA					P	
OLEA	ESR	ARMS	TELEDYNE BROWN	HUNTSVILLE	AL						
OLEA	ESR	ANALYSIS W/S	KNOWLEDGEWARE	ATLANTA	GA					P	
OLEA	ESR		FORT BENNING								
OLEA	ESR		THE ASSOCIATED	NEWPORT	RI						
OLEA	NL	SMART TRANSL	SMART COMM INC	NEW YORK	NY						
OLEA	NL	SMART EXP ED	SMART COMM INC	NEW YORK	NY						
OLEA	NL	PERIPHASE	ALPNET	SALT LAKE CIT	UT	I	V	S		P	
OLEA	NL	NL/PASER	L/TEK INC	DAVIS	CA						
OLEA	NL	NATRL ACCESS	TEXAS INSTRUMEN	AUSTIN	TX						
OLEA	NL	NAT LANG TOO	COGNITIVE SYS	NEW HAVEN	CT		V				
OLEA	NL	LANGUAGECRAF	CARNEGIE GROUP	PITTSBURG	PA	3	V				
OLEA	NL	INTELLECT	AI CORP	WATLTAM	MA	I		S			
OLEA	NNW	NEURO SHELL	WARD SYSTEMS GR	FREDERICK	MD					P	
OLEA	NNW	NEURALWKSPRO	NEURALWARE INC	SEWICKLEY	PA			S		P	
OLEA	NNW	NEURALWKSEXP	NEURALWARE INC	SEWICKLEY	PA			S		P	
OLEA	NNW	NEURAL-NET	INTEC	TEMPE	AZ					P	
OLEA	NNW	NESTOR DEVEL	NESTOR INC	PROVIDENCE	RI			S		P	
OLEA	NNW	N-NET	AI CORP								
OLEA	NNW	N-NET	AI WARE	CLEVELAND	OH		V			P	
OLEA	NNW	ASPN	BARRON ASSOCIAT	STANDARDSVILL	VA		V	S			M
OLEA	NNW	AI NET	AI WARE INC	CLEVELAND	OH					P	
OLEA	NNWST	NN100/NN170	INTEC							P	
OLEA	NNWST	NN-100	INTEC							P	
OLEA	SHELL	X1 RULE	EXPERTEACH INC	INCLINE VILLA	NV					P	
OLEA	SHELL	X1 PLUS	EXPERTEACH INC	INCLINE VILLA	NV					P	

Technology for the Assessment of Software Quality "TASQ"

APPL_N	TYPE	TOOL_NAME	COMPANY	CITY & STATE		IBM	VAX	SUN	APOLLO	IBM	MAC
OLEA	SHELL	VP EXPERT	PAPERBACK S/W	BERKELEY	CA					P	
OLEA	SHELL	VAXOPS5	DEC	HJUNTSVILLE	AL		V				
OLEA	SHELL	VAX DECISION	DEC	HUNTSVILLE	AL		V				
OLEA	SHELL	TOPSI	DYNAMIC MASTER	ATLANTA	GA					P	
OLEA	SHELL	TIMM	GEN RESRCH CORP	MCLEAN	VA	I	V			P	
OLEA	SHELL	SUPEREXPERT	SOFTSYNC INC	NEW YORK	NY						
OLEA	SHELL	STAR	COSMIC	ATHENS	GA		V	S			
OLEA	SHELL	SPE	SUN MICROSYSTEM	MOUNTAINVEW	CA			S			
OLEA	SHELL	SPC	MIT	CAMBRIDGE	MA						
OLEA	SHELL	SIERRA OP55	INFERENCE ENGIN	CAMBRIDGE	MA					P	
OLEA	SHELL	SELECTOR II	CHARTERED ELECT	SAN FRANCISCO	CA					P	
OLEA	SHELL	SD ADVISOR	SYS DESIGN INT	NEW CASTLE	DE						
OLEA	SHELL	RULEMASTER I	RADIAN CORP.	AUSTIN	TX		V	S	A	P	
OLEA	SHELL	RPM	CARNEGIE GROUP	PITTSBURG	PA		V	S			
OLEA	SHELL	PROGENESIS	QUANTUM INKNOWR	SANTA CLARA	CA					P	
OLEA	SHELL	PROCEDURE CO	TEXAS INSTRUMEN	AUSTIN	TX					P	
OLEA	SHELL	OPS5E	BALL SYS ENG DI	SAN DIEGO	CA						
OLEA	SHELL	OPS/83	PROD SYS TECH I	PITTSBURGH	PA		V	S	A		
OLEA	SHELL	NEXPERT OBJE	NEURON DATA	PALO ALTO	CA	I	V	S	A	P	M
OLEA	SHELL	MICRO EXPERT	MCGRAW HILL BOO	NEW YORK	NY					P	
OLEA	SHELL	MAC SMARTS	COGNTRON TECH	CAMBRIDGE	MA						
OLEA	SHELL	LOOPS	ENVOS	ARLINGTON	VA			S			
OLEA	SHELL	LEVEL 5	LEVEL 5 RESEARC	INDIALANTIC	FL	I	V				M
OLEA	SHELL	KNOWLEDGEPRO	KNOWLEDGE GARDE	NASSAU	NY					P	
OLEA	SHELL	KNOWLEDGE CR	CARNEGIE GROUP	PITTSBURGH	PA		V	S			
OLEA	SHELL	KEYSTONE	KEYSTONE TECH	JACKSONVILLE	FL					P	
OLEA	SHELL	KES II	S/W ARCH & ENGI	ARLINGTON	VA		V	S	A	P	
OLEA	SHELL	KES	S/W ARCH. & ENG	ARLINGTON	VA		V	S	A	P	
OLEA	SHELL	KEE	INTELLICORP	IRVING	TX	I	V	S	A		M
OLEA	SHELL	KDS3	KDS CORP.	WILMITTE	IL					P	

Technology for the Assessment of Software Quality "TASQ"

APPL_N	TYPE	TOOL_NAME	COMPANY	CITY & STATE		IBM	VAX	SUN	APOLLO	IBM	MAC
OLEA	SHELL	KDS2	KDS CORP	WILMITTE	IL					P	
OLEA	SHELL	KBVISION	AMERINEX AI	AMHERST	MA						
OLEA	SHELL	KBMS	AI CORP	WALTHAM	MA	I		S			
OLEA	SHELL	JOSHUA	SYMBOLICS	MOUNTAINVIEW	CA						
OLEA	SHELL	INTELLIGENCE	INTELLIGENCE WA	LOS ANGELES	CA					P	
OLEA	SHELL	ICAT	AUTOMATED REASO	ROSLYN	NY		V	S			M
OLEA	SHELL	IBIS PLUS	INTELLIGENCE MF							P	
OLEA	SHELL	IBIS	INTELLIGENCE MF	WEST SACRAMEN	CA					P	
OLEA	SHELL	HUMBLE	XEROX CORP.	PASADENA	CA			S	A	P	M
OLEA	SHELL	GURU	MICRO DATABASE	LAFAYETTE	IN		V	S		P	
OLEA	SHELL	GOLDWORKS	GOLDHILL COMPUT	CAMBRIDGE	MA			S		P	M
OLEA	SHELL	GEST	GA TECH RESEARC	ATLANTA	GA		V				
OLEA	SHELL	GEN-X	GENERAL ELECTRI	SCHENECTADY	NY					P	
OLEA	SHELL	G2	GENSYM CORP	CAMBRIDGE	MA		V	S	A		M
OLEA	SHELL	FORTH EXPERT	MOUNTAIN VIEW	MOUNTAIN VIEW	CA						
OLEA	SHELL	FLOP	KEMP CARAWAY HE	BIRMINGHAM	AL					P	
OLEA	SHELL	EXSYS PROFES	EXSYS INC	ALBURQUERQUE	NM		V	S		P	
OLEA	SHELL	EXSYS	EXSYS INC.	ALBURQUERQUE	NM		V	S		P	
OLEA	SHELL	EXPERT-2	MILLER MICROCOM	NATICK	MA					P	
OLEA	SHELL	EXPERT THINK	TRANSPOWER CORP								
OLEA	SHELL	EXPERT SYS	COSMIC	ATHENS	GA						
OLEA	SHELL	EXPERT R	COYNE KALA INC	ARLINTON	VA						
OLEA	SHELL	EXPERT EDGE	JEFFREY PERRON	SAN FRANCISCO	CA					P	
OLEA	SHELL	EXPERT EASE	JEFFREY PERRON	SAN FRANCISCO	CA					P	
OLEA	SHELL	EXPERT CONTR	UMECORP	LARKSPUR	CA						
OLEA	SHELL	EXPERT CHOIC	DECISION SUPPOR	MCLEAN	VA					P	
OLEA	SHELL	EXPERT 87	MAGIC 7 S/W	LOS ALTOS	CA					P	
OLEA	SHELL	EXPER OPS5+	EXPERTELLIGENCE	SANTA BARBARA	CA						M
OLEA	SHELL	EXPER OPS5	EXPERTELLIGENCE	SANTA BARBARA	CA						M
OLEA	SHELL	EXPER FACTS	EXPERTELLIGENCE	SANTA BARBARA	CA						M

Technology for the Assessment of Software Quality "TASQ"

APPL_N	TYPE	TOOL_NAME	COMPANY	CITY & STATE		IBM	VAX	SUN	APOLLO	IBM	MAC
OLEA	SHELL	EXADS	COSMIC	ATHENS	GA		V			P	
OLEA	SHELL	EST	MIND PATH TECH	DALLAS	TX					P	
OLEA	SHELL	ESP FRAME EN	EXPERT SYS INTE	PHILADELPHIA	PA					P	
OLEA	SHELL	ELOQUENT	ELOQUENT SYS CO	MANCHESTER	NH						M
OLEA	SHELL	EASY EXPERT	PARK ROW S/W	SAN DIEGO	CA					P	
OLEA	SHELL	DEV 2.0	LEVEL 5 RESEARC	INDIALANTIC	FL	I	V				M
OLEA	SHELL	COGNATE	PERIDOM INC	BOWIE	MD						M
OLEA	SHELL	CLIPS	COSMIC	ATHENS	GA		V			P	M
OLEA	SHELL	CERBERUS	COSMIC	ATHENS	GA		V				
OLEA	SHELL	C EXPERT	S/W PLUS	CROFTON	MD					P	
OLEA	SHELL	ART	INFERENCE CO.	LOS ANGELES	CA	I	V	S	A	P	
OLEA	SHELL	ARITY/ES	ARITY CORP.	CONCORD	MA					P	
OLEA	SHELL	APPL EXPERT	CULLINET S/W IN	HOUSTON	TX	I	V				
OLEA	SHELL	ALEX	HARRIS & HALL	PORT ANGELES	WA					P	
OLEA	SHELL	AESOP EXPERT	COSMIC	ATHENS	GA		V				
OLEA	SHELL	AES	AION DEV SYSTEM	DALLLAS	TX	I				P	
OLEA	SHELL	ADVISOR 2	EXPERT SYS INTE	PHILADELPHIA	PA		V			P	
OLEA	SHELL	ADS	AION DEV SYSTEM	DALLAS	TX	I				P	
OLEA	SHELL	1XL	INTELLIGENCE WA	LOS ANGELES	CA					P	
OLEA	SHELL	1ST CLASS FU	1ST CLASS EXPER	WAYLAND	MA					P	
OLEA	SHELL	1ST CLASS	1ST CLASS EXPER	WAYLAND	MA					P	
OLEA	SST	TABLET	C A INTELLEGEN	SAN FRANCISCO	CA		V	S		P	
OLEA	SST	SIMKIT	INTELLICORP	IRVING	TX	I	V	S	A	P	M
OLEA	SST	ROOMS	ENVOS	ARLINGTON	VA			S			
OLEA	SST	REALM	KEYSTONE TECH	JACKSONVILLE	FL						
OLEA	SST	PLEXI	SYMBOLICS	CAMBRIDGE	MA						
OLEA	SST	NOTECARD	ENVOS	ARLINGTON	VA			S			
OLEA	SST	MEDLEY	ENVOS	ARLINGTON	VA			S			
OLEA	SST	KNW PRO GRPH	KNOWLEDGE GARDE							P	
OLEA	SST	KNOWLEDGEMAK	KNOWLEDGE GARDE	NASSAU	NY					P	

Technology for the Assessment of Software Quality "TASQ"

APPL_N	TYPE	TOOL_NAME	COMPANY	CITY & STATE		IBM	VAX	SUN	APOLLO	IBM	MAC
OLEA	SST	KNOWLEDGE DB	KNOWLEDGE GARDE	NASSAU	NY					P	
OLEA	SST	KEE CONNECTI	INTELLICORP	IRVING	TX	I	V	S		P	M
OLEA	SST	HPO	AION DEV SYSTEM	DALLAS	TX	I				P	
OLEA	SST	GRAPHICS	C A INTELLEGEN				V	S		P	
OLEA	SST	FRAME/DBFRAM	C A INTELLEGEN	SAN FRANCISCO	CA		V	S		P	
OLEA	SST	EXPERTEACHII	INTELLIGENCE WA	LOS ANGELES	CA					P	
OLEA	SST	EXCHANGE	C A INTELLIGEN	SAN FRANCISCO	CA		V	S		P	
OLEA	SST	EX123	C A INTELLEGEN				V	S		P	
OLEA	SST	ES STARTER	MIND PATH TECH	DALLAS	TX					P	
OLEA	SST	CONSULTANT	C A INTELLIGEN	SAN FRANCISCO	CA		V	S		P	
OLEA	SST	CONSENUS BLD	MAGIC 7 S/W	LOS ALTOS	CA					P	
OLEA	SST	AUTO INTELLI	INTELLIGENCE WA	LOS ANGELES	CA					P	

APPENDIX D

CLASSSIFICATION TOOLS MATRIX

[A single copy of the composite classification matrix used to subsequently categorize the tools/environment of Appendix C is separately bound]